GEMSTONE®

GEMFIRE®

ENTERPRISE

GemFire Enterprise® Monitor User's Manual

Version 2

December 2008

Send comments on this guide to docs@gemstone.com

INTELLECTUAL PROPERTY OWNERSHIP

This documentation is furnished for informational use only and is subject to change without notice. GemStone Systems, Inc. assumes no responsibility or liability for any errors or inaccuracies that may appear in this documentation.

This documentation, or any part of it, may not be reproduced, displayed, photocopied, transmitted, or otherwise copied in any form or by any means now known or later developed, such as electronic, optical, or mechanical means, without express written authorization from GemStone Systems, Inc.

Warning: This computer program and its documentation are protected by copyright law and international treaties. Any unauthorized copying or distribution of this program, its documentation, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted under the maximum extent possible under the law.

The software installed in accordance with this documentation is copyrighted and licensed by GemStone Systems, Inc. under separate license agreement. This software may only be used pursuant to the terms and conditions of such license agreement. Any other use may be a violation of law.

Use, duplication, or disclosure by the Government is subject to restrictions set forth in the Commercial Software - Restricted Rights clause at 52.227-19 of the Federal Acquisitions Regulations (48 CFR 52.227-19) except that the government agency shall not have the right to disclose this software to support service contractors or their subcontractors without the prior written consent of GemStone Systems, Inc.

This software is provided by GemStone Systems, Inc. and contributors “as is” and any expressed or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall GemStone Systems, Inc. or any contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

COPYRIGHTS

This software product, its documentation, and its user interface Copyright (c) 1997-2008, GemStone Systems, Inc. All Rights Reserved by GemStone Systems Inc.

The Java Software technologies are Copyright (c) 1994-2000 Sun Microsystems, Inc. Copyright (c) Eclipse Contributors and others 2000, 2006. Copyright 1999-2006 The Apache Software Foundation. ILOG JViews50 copyright 1996-2001 ILOG S.A. All rights reserved.

Eclipse v3.2.2 Copyright (c) Eclipse Contributors and others 2000, 2006.

Xerces 2.9.0 Copyright (c) 1999-2006 The Apache Software Foundation.

ILOG JViews 5.0 ILOG JViews50 copyright 1996-2001 ILOG S.A. All rights reserved.

PATENTS

GemFire is protected by U.S. patent 6,360,219. Additional patents pending.

TRADEMARKS

GemStone, GFMon, GemFire, GemFire Enterprise, and the GemStone logo are trademarks or registered trademarks of GemStone Systems, Inc. in the United States and other countries (trademark application pending for GemFire).

UNIX is a registered trademark of The Open Group in the U. S. and other countries.

Linux is a registered trademark of Linus Torvalds.

Red Hat and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.

SUSE is a registered trademark of SUSE AG.

Sun, Sun Microsystems, Solaris, Forte, Java, Java Runtime Edition, JRE, and other Java-related marks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

Intel and Pentium are registered trademarks of Intel Corporation in the United States and other countries.

Microsoft, Windows, and Visual C++ are registered trademarks of Microsoft Corporation in the United States and other countries.

W3C is a registered trademark of the World Wide Web Consortium.

Other company or product names mentioned herein may be trademarks or registered trademarks of their respective owners. Trademark specifications are subject to change without notice. All terms mentioned in this documentation that are known to be trademarks or service marks have been appropriately capitalized to the best of our knowledge; however, GemStone cannot attest to the accuracy of all trademark information. Use of a term in this documentation should not be regarded as affecting the validity of any trademark or service mark.

GemStone Systems, Inc.

1260 NW Waterhouse Avenue, Suite 200
Beaverton, OR 97006

Table of Contents

Preface ***9***

How This Documentation Is Organized 9
Typographical Conventions 9
Other Useful Documents 10
Preserving Artifacts for Technical Support 11
Contacting Technical Support 12
24x7 Emergency Technical Support 12
GemStone Support Web Site 13

Chapter 1. Product Installation ***15***

1.1 System Requirements 16
Windows 16
Linux 16
Java Runtime Requirements 16
Other Software Requirements 16
Runtime Requirements 17
1.2 Installing and Uninstalling GemFire Enterprise Monitor 18
1.3 GFMon License Files 19
1.4 GFMon Product Tree 20
GFMon Product Documentation 20

Chapter 2. System Configuration ***21***

Starting an Admin Agent Using Locators 22
Starting an Admin Agent Using Multicast 22
Running Multiple Instances of GFMon 23
Disconnecting From the Agent in GFMon 23

Chapter 3. Accessing GemFire System Information Using GFMon ***25***

3.1 Accessing Information 26
Left Panel 26
Overview Panel 27

Data Panel 31
Members Panel 33
Alerts Panel 35
Custom Charts Panel 38
Preferences Panel 45
Status Bar 47
Disconnecting From the Agent in GFMon 48
3.2 Error Notifications Displayed by GFMon 49

Glossary ***51***

Index ***61***

List of Figures

Figure 3.1	Left Panel	26
Figure 3.2	Overview Panel	27
Figure 3.3	Details for Member Panel	29
Figure 3.4	Data Region Panel	31
Figure 3.5	Members Panel	33
Figure 3.6	Alerts Panel	35
Figure 3.7	Details for Events in the GFMon Event Viewer	36
Figure 3.8	Custom Charts — Chart Configuration Tab	38
Figure 3.9	Using Filters in the Custom Charts Panel	40
Figure 3.10	Garbage Collection Statistics Chart	41
Figure 3.11	Cache Hit Ratios Chart	42
Figure 3.12	Used Memory Usage Chart	42
Figure 3.13	Active CQ Queries Chart	43
Figure 3.14	Response Times Chart	43
Figure 3.15	Cache Entry Count Chart	44
Figure 3.16	CQ Query Times Chart	44
Figure 3.17	Preferences — Connections Tab	45
Figure 3.18	Preferences — General Tab	46
Figure 3.19	Status Bar	47
Figure 3.20	The Progress View	48

List of Tables

Table 1.1	Operating System Support Matrix	16
Table 1.2	Runtime Requirements	17
Table 3.1	Event types Logged in the GFMon Event Viewer	37

Preface

About This Guide

This guide describes the GemStone® GemFire Enterprise® Monitor (GFMon™) functions that monitor the GemFire Enterprise product, which is typically used for distributed caching and data distribution. For information about the GemFire Enterprise product, see the *GemFire™ Enterprise Documentation* located on www.gemstone.com.

How This Documentation Is Organized

This guide contains the following sections:

- Chapter 1, *Product Installation*, on page 15 lists system requirements and tells how to install GFMon.
- Chapter 2, *System Configuration*, on page 21 tells how to set up your GemFire Enterprise Monitor system.
- Chapter 3, *Accessing GemFire System Information Using GFMon*, on page 25 describes the functionality of various panels in GFMon.
- *Glossary* on page 51 provides a list of terms and their definitions.

Typographical Conventions

This document uses the following typographical conventions:

- Methods, types, file names and paths, folder names, code listings, and prompts are shown in `Courier New` typeface. For example:

  ```
  gfPut
  ```

- Parameters and variables are shown in *`italic`* font. For example,

  ```
  gfConnect(sysDir, connectionName, writeProtectAllowed)
  ```

- In examples showing both user input and system output, the lines you type are distinguished from system output by **`boldface`** type:

  ```
  prompt> gemfire
  ```

- If you are viewing this document online, the page, section, and chapter references are hyperlinks, like this reference to *Installing and Uninstalling GemFire Enterprise Monitor* on page 18 and this reference to Chapter 3, *Accessing GemFire System Information Using GFMon*, on page 25. Blue text denotes a hyperlink.

Other Useful Documents

The *GemFire Enterprise online Java API documentation* can be accessed through the file `index.html` in the GemFire's `docs` directory. This file is located under your Gemfire (GFE) installation as `docs/japi`.

The *GemFire Enterprise Developer's Guide* describes the major services and functions in GemFire Enterprise.

The *GemFire Enterprise System Administrator's Guide* describes the system administration functions required to install, deploy, and manage the GemFire Enterprise product.

The *Visual Statistics Display* guide describes how to use the Visual Statistics Display (VSD) tool, which is used to analyze archived historical data. Contact GemStone Technical Support for instructions about acquiring VSD and its documentation.

Technical Support

GemStone provides several sources for product information and support. The *GemFire Enterprise Developer's Guide*, the *GemFire Enterprise System Administrator's Guide* and the GemFire Enterprise online Java API provide extensive documentation, and should always be your first source of information for GemFire Enterprise. The current document provides functional information about GemFire Enterprise Monitor (GFMon™). GemStone Technical Support engineers will refer you to these documents when applicable. However, you may need to contact Technical Support for the following reasons:

- Your technical question is not answered in the documentation.
- You receive an error message that directs you to contact GemStone Technical Support.
- You want to report a bug.
- You want to submit a feature request.

Questions concerning product availability, pricing, license keyfiles, or future features should be directed to your GemStone account manager.

Preserving Artifacts for Technical Support

Before you call Support, save all the artifacts including:

- Log files - `GFMon.log` located in the Log File directory. To customize the log file location, see *Preferences Panel* on page 45. The default location of the log file is `<GFMon product Dir>/logs/<user name>` directory. Send the full log to Technical Support, not just the stack. Even at the default logging level, the log contains data that may be important, such as the operating system and license.
- All the contents of the `workspace` directory in the `<GFMon product directory>` folder.
- For Windows, save the Dr. Watson output. This is located in the `C:\Documents and Settings\All Users\Application Data\Microsoft\Dr Watson` directory.

  You can configure Dr. Watson as the default debugger by typing the **`drwtsn32 -I`** *command from a Windows command prompt.*

- Process core dump files and crash dump files on UNIX systems.
- For Linux, you can use `gdb` to extract a stack from a core dump file. Call Technical Support if you need assistance.

If the GFMon application on your virtual machine (VM) is hung and you do not have to close it, retain the application while you contact Technical Support. If you cannot leave the VM running, and it is running under Unix or Cygwin on Microsoft® Windows®, signal it twice with this command, letting five to ten seconds pass between the two signals:

```
kill -QUIT pid
```

This sends the stack dumps of the VM to the log file for inspection. For Windows systems, call Technical Support for assistance in obtaining stack dumps.

Do not delete any files until you call Technical Support.

Contacting Technical Support

When contacting GemStone Technical Support, please be prepared to provide the following information:

- Your name, company name, and GemFire license number
- The GFMon product version you are using
- The hardware platform and operating system you are using
- A description of the problem or request
- Exact error messages received, if any
- Any artifacts in the preceding list

Your GemStone support agreement may identify specific individuals who are responsible for submitting all support requests to GemStone. If so, please submit your information through those individuals. All responses will be sent to authorized contacts only.

For non-emergency requests, you should contact Technical Support by web form or E-mail. You will receive confirmation of your request, and a request assignment number for tracking. Replies will be sent by E-mail whenever possible, regardless of how they were received.

GemStone Support Website: http://support.gemstone.com

This is the preferred method of contact. The Help Request link is at the top right corner of the home page—please use this to submit help requests. This form requires an account, but registration is free of charge. To get an account, just complete the Registration Form, found in the same location. You will be able to access the site as soon as you submit the web form.

E-mail: mailto:support@gemstone.com

Please do not send files larger than 100K (for example, core dumps) to this address. A special address for large files will be provided as appropriate.

Telephone: (800) 243-4772 or (503) 533-3503

We recommend that you call only for more serious requests that require immediate evaluation, such as a production system that is non-operational.

Emergency requests are handled by the first available engineer. If you are reporting an emergency and you receive a recorded message, do not use the voicemail option. Transfer your call to the operator, who will take a message and immediately contact an engineer.

Non-emergency requests received by telephone are placed in the normal support queue for evaluation and response.

24x7 Emergency Technical Support

GemStone offers, at an additional charge, 24x7 emergency technical support. This support entitles customers to contact us 24 hours a day, 7 days a week, 365 days a year, if they encounter problems that cause their production application to go down, or that have the potential to bring their production application down. Contact your GemStone account manager for more details.

GemStone Support Web Site

The GemStone support web site, at http://support.gemstone.com, provides a variety of information to help you use GemStone products. Use of this site requires an account, but registration is free of charge. To get an account, just complete the Registration Form, found in the same location. You'll be able to access the site as soon as you submit the web form.

The following types of information are provided at this web site:

- **Help Request** is an online form that allows designated technical support contacts to submit requests for information or assistance via E-mail to GemStone Technical Support.
- **Technotes** provide answers to questions of general interest submitted by GemStone customers. They may contain coding examples, links to other sources of information, or downloadable code.
- **Bugnotes** identify performance issues or error conditions that you may encounter when using a GemStone product. A bugnote describes the cause of the condition, and, when possible, provides an alternative means of accomplishing the task. In addition, bugnotes identify whether a fix is available, either by upgrading to another version of the product, or by applying a patch. Bugnotes are updated regularly.
- **Patches** provide code fixes and enhancements that have been developed after product release. A patch generally addresses a specific group of behavior or performance issues. Most patches listed on the GemStone Web site are available for direct downloading.
- **Tips and Examples** provide information and instructions for topics that usually relate to more effective or efficient use of GemStone products. Some Tips may contain code that can be downloaded for use at your site.
- **Release Notes** and **User's Manual** for your product software are provided in PDF format.
- **Community Links** provide customer forums for discussion of GemStone product issues.

Technical information on the GemStone web site is reviewed and updated regularly. We recommend that you check this site on a regular basis to obtain the latest technical information for GemStone products. We also welcome suggestions and ideas for improving and expanding our site to better serve you.

Training and Consulting

Consulting and training for all GemStone products are available through GemStone's Professional Services organization.

- Training courses are offered periodically at GemStone's offices in Beaverton, Oregon, or you can arrange for onsite training at your desired location.
- Customized consulting services can help you make the best use of GemStone products in your business environment.

Contact your GemStone account representative for more details or to obtain consulting services.

Chapter 1

Product Installation

This chapter covers system requirements and licensing for GemFire Enterprise® Monitor (GFMon™) and describes the product tree. Systems that meet the requirements described here are suitable for installing GFMon and beginning development, but additional system resources may be necessary to support large applications.

This chapter covers the following:

- System Requirements (page 16)
- Installing and Uninstalling GemFire Enterprise Monitor (page 18)
- GFMon License Files (page 19)
- GFMon Product Tree (page 20)

> *If you are upgrading an existing GemFire Enterprise Monitor installation, be sure to check the Release Notes for upgrade and migration instructions.*

To contact GemStone Technical Support:

- On the web: http://support.gemstone.com
- By e-mail: support@gemstone.com
- By phone: 800/243-4772 or 503/533-3503

1.1 System Requirements

This version of GFMon is designed to run with full capabilities on platforms running Linux and Microsoft® Windows®.

Supported Platforms

This section lists the platforms that GFMon supports. If you are interested in a platform that is not listed, please contact your GemStone sales representative. GemStone will evaluate whether it can support the platform, either as-is or under a special agreement, or whether there are reasons that prevent deploying on a particular platform.

GFMon supports the following configurations:

Table 1.1 Operating System Support Matrix

Operating System	JDKs and JREs	
	Sun® JSE 1.5.0_16	Sun JSE 1.6.0_03
Red Hat® Enterprise Linux® 4 (patch 7), kernel version 2.6.9-78 EL, with GIMP Tool Kit (GTK 2.4.13-24.el4)	Yes	No
Windows XP SP3	Yes	Yes
Microsoft Windows Server® 2003 SP2	Yes	Yes

These sections list the operating systems used for building and testing GFMon. GFMon may work with later system releases upon which it has not been tested.

Windows

GemFire Enterprise Monitor has been tested successfully on:

- Windows Server 2003 SP2
- Windows XP Professional SP3

Linux

GFMon is tested on the following Linux version:

- Red Hat EL 4 (patch 7), kernel version 2.6.9-78 EL

If you are not sure of the kernel version on your system, use this command to list it:

```
prompt> uname -r
```

Java Runtime Requirements

A Java Runtime Environment (JRE) 1.5.0 or later must be installed for GFMon to run. GFMon does not include a bundled JRE or JDK so you can choose the version that best suits your unique system requirements. You can download the appropriate JDK or JRE from http://java.sun.com.

Other Software Requirements

Adobe Acrobat Reader is required for viewing the documentation distributed as PDF files. Download a free copy at http://www.adobe.com/products/acrobat/readstep.html.

Runtime Requirements

GFMon has machine-related runtime requirements which are covered in this section:

Table 1.2 Runtime Requirements

Hardware/Software Requirement	Version
Operating System	See *Operating System Support Matrix*
Processor	Intel® 2 GHz Pentium® 4 (or equivalent)
System memory	1 GB RAM (recommended)
Software on the system on which the Admin Agent is configured	GemFire Enterprise 5.7 or later

1.2 Installing and Uninstalling GemFire Enterprise Monitor

GFMon is distributed as a ZIP archive. Extract the ZIP archive to a location in the file system to install the software.

The `gfmon.bat` and `gfmon` files are created in this location. Use these files to run GFMon on Microsoft Windows and Linux, respectively.

> *By default,* `gfmon.bat` *and* `gfmon` *files use the environment variable* `java` *to point to the JDK version that is available in the system. If you want to execute GFMon using a specific version of JDK, you can specify the location of the java executable file using the environment variable* `GF_JAVA`*.*
>
> *Set the environment variable GEMFIRE to point to the GemFire product directory. This directory must be the same as the GemFire product directory which is being used by the JMX admin agent that GFMon connects to monitor the distributed system. The GFMon start-up scripts use the gemfire.jar in the* `GEMFIRE/lib` *directory to launch GFMon.*

You can obtain the instruction file from your GemStone salesperson or from the GemStone website at http://www.gemstone.com/download. To access the website, you need to enter your login and password. If you are new to GemStone products, to register and create a login and password to get into the download center.

You can uninstall GemFire Enterprise Monitor by simply deleting the entire product tree.

1.3 GFMon License Files

GFMon is licensed as a package along with GemStone® GemFire Enterprise®. You do not require a separate license file to run the GFMon application.

However, to run an Admin Agent on GemFire, you require a GemFire license. For more information about GemFire licenses, see the *GemFire Enterprise Documentation*. For licenses, contact keyfiles@gemstone.com.

1.4 GFMon Product Tree

Directory Name	Contents
`configuration`	Configuration files required by GFMon. Also contains run-time plug-ins and log files
`docs`	Contains this user manual
`plugins`	The plug-ins required for the GFMon application
`gfmon.bat`	Batch file that runs GFMon in Windows operating systems
`gfmon`	Script file that runs GFMon in Linux operating systems
`workspace`	This folder is created after you launch the GFMon application. Contains the log files and application metadata.

GFMon Product Documentation

The documentation for your GFMon products is provided in PDF format and can be viewed with any frames-capable web browser. These guides can be accessed through the index page in the GFMon `docs` directory:

Windows	*productDir*`\docs\UserManual_GFMon.pdf`
Linux	*productDir*`/docs/UserManual_GFMon.pdf`

In addition to the current document, the *Release Notes* placed in the GFMon product directory describe differences between the current release and previous product releases.

Chapter

2 System Configuration

The chapter describes system configuration required to deploy GemFire Enterprise® Monitor (GFMon™).

Before you begin with the procedure in this chapter, configure the distributed system with instructions provided in the *GemFire Enterprise® Quickstart Guide*. You can access the GemFire documentation through the `<GemFire product directory>/docs/japi/index.html`.

Ensure that the distributed system has a running Admin Java Management Extensions (JMX) agent with RMI enabled.

Setting Up a Cache Server

1. Start the cache server. To specify the cache server configuration properties during start-up, you can use the Java command line, API, or the `gemfire.properties` file.

   For more information about configuring the cache server, see *Configuring the Cache* in the *GemFire Enterprise® Developer's Guide*.

2. After you have configured the cache server, identify the locator or the multicast information through which membership discovery should occur.

   You can start the cache server by specifying the multicast port number or the locator connection string through the command line or through the `gemfire.properties` file. To set up the server multicast (mcast) port, use the property `mcast-port`. To set up the locator connection string, use the property `locators` which is a comma-delimited `host[port]` string.

   > *When you are using locators to connect, set the* `mcast-port` *to* `0`.

   For information about discovering members in a distributed system see *Member Discovery* in the *GemFire Enterprise® System Administrator's Guide Version 5.7*.

3. The default `gemfire.properties` specification is overridden by the `gfTest` file specification. In the example below, the command-line specification for `mcast-port` overrides any setting that might be provided in the `gfTest` file or in any `Properties` object:

   ```
   java -DgemfirePropertyFile=gfTest -Dgemfire.mcast-port=10999 test.Program
   ```

   In the example below, the command-line specification for `locators` overrides any setting that might be provided in the `gfTest` file or in any `Properties` object:

   ```
   java -DgemfirePropertyFile=gfTest -Dgemfire.locators=bishop[10334]
   test.Program -Dgemfire.mcast-port=0
   ```

Configuring an Admin Agent

After you have configured a distributed system, configure an agent to connect to the distributed system. To configure an Admin Agent see *Using JMX to Administer GemFire* in the *GemFire Enterprise® System Administrator's Guide*.

1. In the agent configuration, the `rmi-enabled` parameter must be set to `true`. Start the agent, specifying the RMI listen port and RMI-enabled parameter as part of the start-up arguments or as part of the `agent.properties` file.

   ```
   agent start rmi-enabled=true rmi-port=1199
   ```

2. Record the agent host-name and the RMI-port number from the `agent.log` file. To configure the RMI-port to any valid port, set `rmi-port` in the `agent.properties` file to a valid port number.

   The Agent uses a default port of 1099 as the listener port for incoming RMI connections from JMX clients including GFMon. If the default port 1099 is not available, the Agent chooses the next available port. However, if an RMI-port is specified (either through the command line or via `agent.properties`*) and the port is not available, the JMX Agent reports an error during start-up.*

3. You can verify the RMI-port value from the `agent.log` file which is created once the agent starts

   To verify the host name and port number that you will need to specify in GFMon, you can also list the active connections using this command: `netstat-a`

After you start the GemFire JMX agent, use the agent host name and the RMI port of the agent to connect to the distributed system.

Starting an Admin Agent Using Locators

To start the agent using locators, specify the connection string of the locator. This can be a comma-delimited host[port] list specified in the `agent.properties` file or on the command line as part of the start-up arguments.

```
agent start locators=bishop[10334] mcast-port=0
```

When you are using locators to connect, set the `mcast-port` to `0`.

Starting an Admin Agent Using Multicast

To start the agent using multicast, specify the `mcast-port` connection string. This can be specified in `agent.properties` file or on the command line as part of the start-up arguments.

```
agent start mcast-port=10999
```

Starting GFMon and Connecting to the Admin Agent

To start GFMon:

1. Set the environment variable GEMFIRE to the `<GemFire product directory>`. This directory must be the same as the GemFire product directory which is being used by the JMX admin agent that GFMon connects to monitor the distributed system. The GFMon start-up scripts use the `gemfire.jar` in the `GEMFIRE/lib` directory to launch GFMon.
2. Use `gfmon.bat` to launch GFMon on Microsoft® Windows® platforms and `gfmon` to launch GFMon on Linux platforms.

To connect to the admin agent:

1. In the **Quick Connect** section, enter the host name of the host on which the Admin Agent is running, and the RMI-port number for the agent that the netstat reported or use the information in the `agent.log` file.
2. Click **Connect** to initiate the connection.

Running Multiple Instances of GFMon

To run multiple instances of GFMon, change the log file directory for each instance while launching the tool. In the environment variable `JAVA_ARGS`, the parameter `-Dcom.gemstone.gemfire.tools.prefs.Defaults.prefs_dir` must be set to a different directory than the current logging directory. For example:

```
set JAVA_ARGS= -Dcom.gemstone.gemfire.tools.prefs.Defaults.prefs_dir=
D:/logs/instance1
```

Disconnecting From the Agent in GFMon

To disconnect from the agent that is running on the GemFire Enterprise distributed system, simply click **Disconnect** on the left panel of the user interface.

Chapter 3

Accessing GemFire System Information Using GFMon

GemFire Enterprise® Monitor (GFMon™) is an application that manages and monitors a GemFire Enterprise distributed system in real-time. GFMon allows you to obtain and view the following information:

- Overall health indicators of the distributed system
- Aggregate and detailed information on various operational and configurational aspects of the distributed system
- System alerts that allow you to take the appropriate corrective action
- Overall system information
- Vital statistics of all members in the distributed system including health statistics for clients
- Summary of alerts that have occurred across the distributed system

GFMon also provides the following features:

- Ability to create custom alerts by specifying system statistic thresholds.
- A rich graphical user interface (GUI) with information on memory and CPU usage across the servers in the distributed system.

GFMon uses JMX to retrieve information from the distributed system.

This chapter provides details about features and functionality of the various panels in GemFire Enterprise Monitor. These topics are covered in detail in the rest of the chapter.

- *Left Panel*
- *Overview Panel*
- *Data Panel*
- *Members Panel*
- *Alerts Panel*
- *Custom Charts Panel*
- *Preferences Panel*
- *Status Bar*

3.1 Accessing Information

The following sections provide information on accessing real-time information from a distributed system running Gemfire Enterprise and setting GFMon preferences.

Left Panel

Click the specific option in the GFMon section to navigate across the various information panels.

Figure 3.1 Left Panel

Use the **Quick Connect** section to specify the **Agent Host** and the **Agent Port** to which the connection must be established. The **System Status** section displays the state of the distributed system to which you are connected. The following colors are displayed:

- Green - the system is functioning normally
- Yellow - user-defined statistic alerts are raised and/or system alerts with severity *Warning* are raised.
- Red - system alerts with severity *Error* or *Severe* are raised and/or a member has crashed.

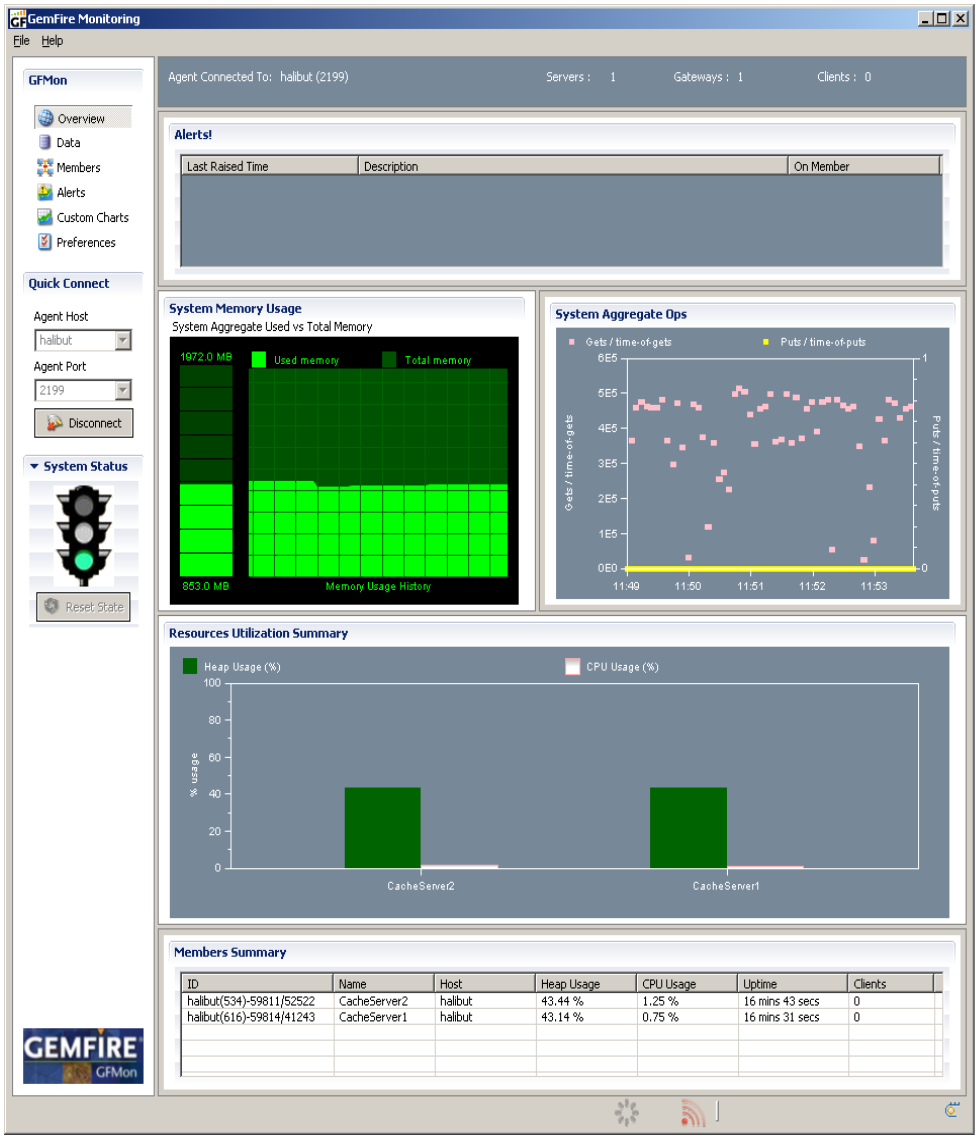
To reset the GFMon **System Status** indicator, click **Reset State**. For more information, point the mouse-pointer at any lights displayed in the **System Status** section.

Tool-tips: When you place the mouse pointer over various GFMon user interface (UI) elements, tool-tips appear and provide to you additional information about the specific UI element.

Overview Panel

The **Overview** panel provides the aggregated system health information of all members in the distributed system, the agent that you are connected to, summary view of alerts from all members in the distributed system, and basic member information.

Figure 3.2 Overview Panel



- Alerts! - The **Alerts!** table displays a summary of alerts triggered by events occurring in the distributed system. The table displays both System alerts and the Statistics alerts defined by the user.

  System alerts include log entries that are tagged as *Severe*, *Error*, or *Warning*. System alerts also consist of alerts indicating unexpected disconnection of a member from the distributed system, unexpected disconnection of the Admin Agent, and errors occurring in GFMon.

Alerts are aggregated, that is, the event triggering the alert is shown as a single entry with updates only in the **Last Raised Time** column. Distinct Statistic alerts defined by the user are aggregated for each member. System alerts are aggregated across the members by the type of event that triggers the alert. The color in the **System Status** section in the Left Panel changes to yellow or red on occurrence of alerts.

To clear an alert in the **Alerts!** section, right-click and select **Clear**.

- You may reset this to green to acknowledge that the alerts have been examined. On reset, the aggregation of the existing alert entries in the table is stopped and they are greyed out. New alerts, even if they are due to events that triggered earlier alerts get displayed as new rows in the table. You may still click on the greyed rows to view their details.
- The maximum number of rows in this and other tables in GFMon can be configured from the Preferences Panel.
- System Memory Usage - This section displays the aggregate memory used across all active servers. The animated graph renders the memory used as a percentage of the total available memory across the servers over time. The vertical bar on the left shows a snapshot of the current memory usage.

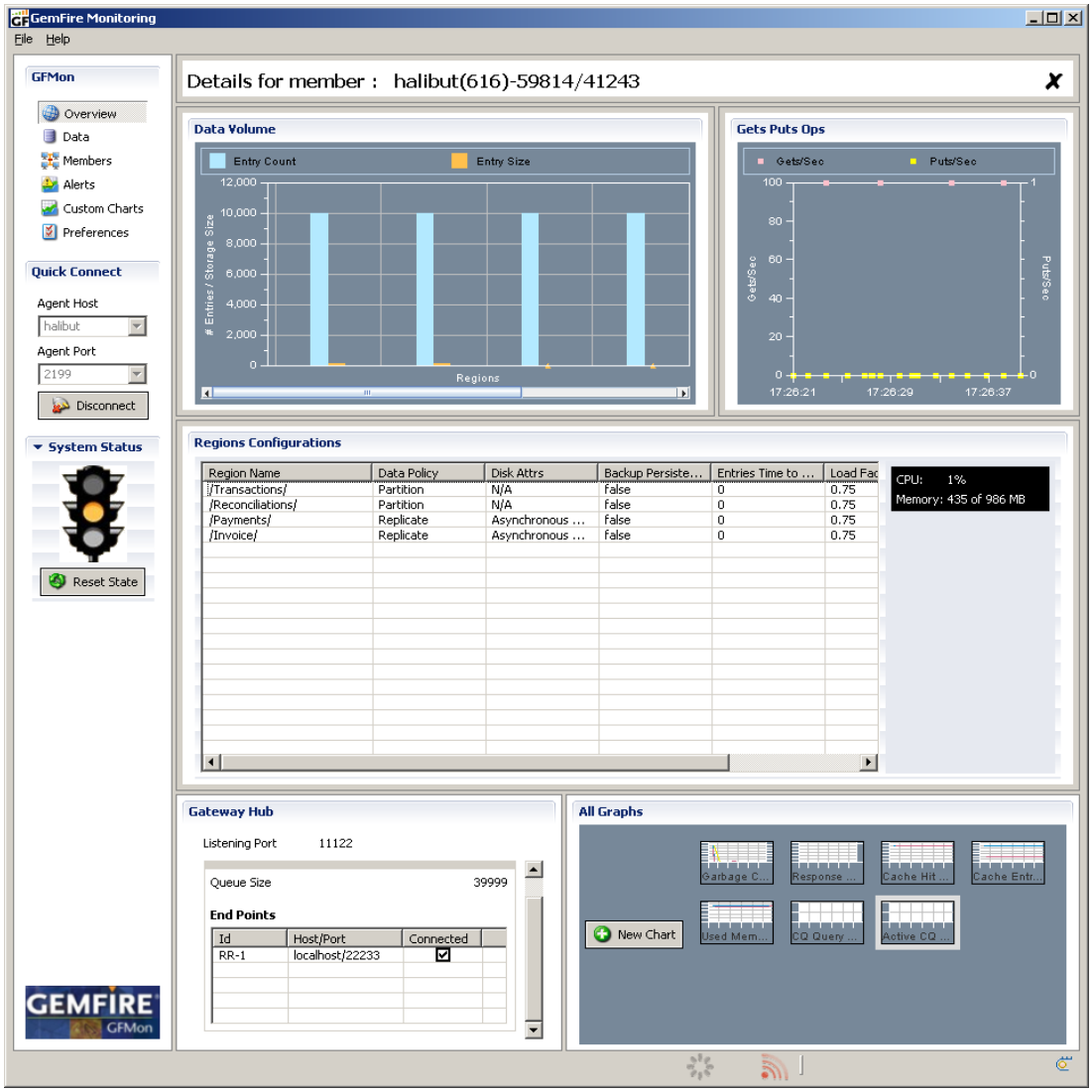
  If the System Memory Usage data is not available, then the X-axis is labeled as **Heap Memory Data is not Available**. If the System Memory Usage data is available, then the X-axis is labeled as **Memory Usage History**.
- System Aggregate Ops - This section displays the aggregate information of the system as a scatter graph in puts/sec and gets/sec.

  *If the time statistics are not enabled, the X-axis is labeled as **Gets-Time, Puts-Time not Available. Time Statistics are not enabled** and no data is displayed. Enable time statistics in your GemFire distribute system to view the data with scale on the X-axis.*
- Resource Utilization Summary - This section provides an animated bar chart of individual memory and CPU usage on each member in the distributed system. The CPU usage bar and memory bar for each node are grouped together.
- Members Summary - This section displays a tabular view of vital member statistics like:
  - ID- the Member ID is a unique auto-generated value that identifies the member in the distributed system.
  - Name - the Name of the member. You can provide this name while starting GemFire through the `gemfire.properties` file or the name is inferred from the member ID.
  - Host - the host on which the member is running.
  - Heap Usage - displays the used heap memory of the member VM as a percentage of the maximum heap memory.
  - CPU Usage - displays the percentage of the process CPU uptime.
  - Clients - number of clients in the cache server if applicable.
  - Uptime - total time the node has been up and running for.

  *Each member in the **Members Summary** section displays the member ID. Other information panels display the member name. If you do not configure a member name, GFMon constructs the member name from the member ID.*

Double-click on a row in the **Members Summary** section to open the **Details for member** screen. The **Details for member** screen provides additional information about each member.

Figure 3.3 Details for Member Panel

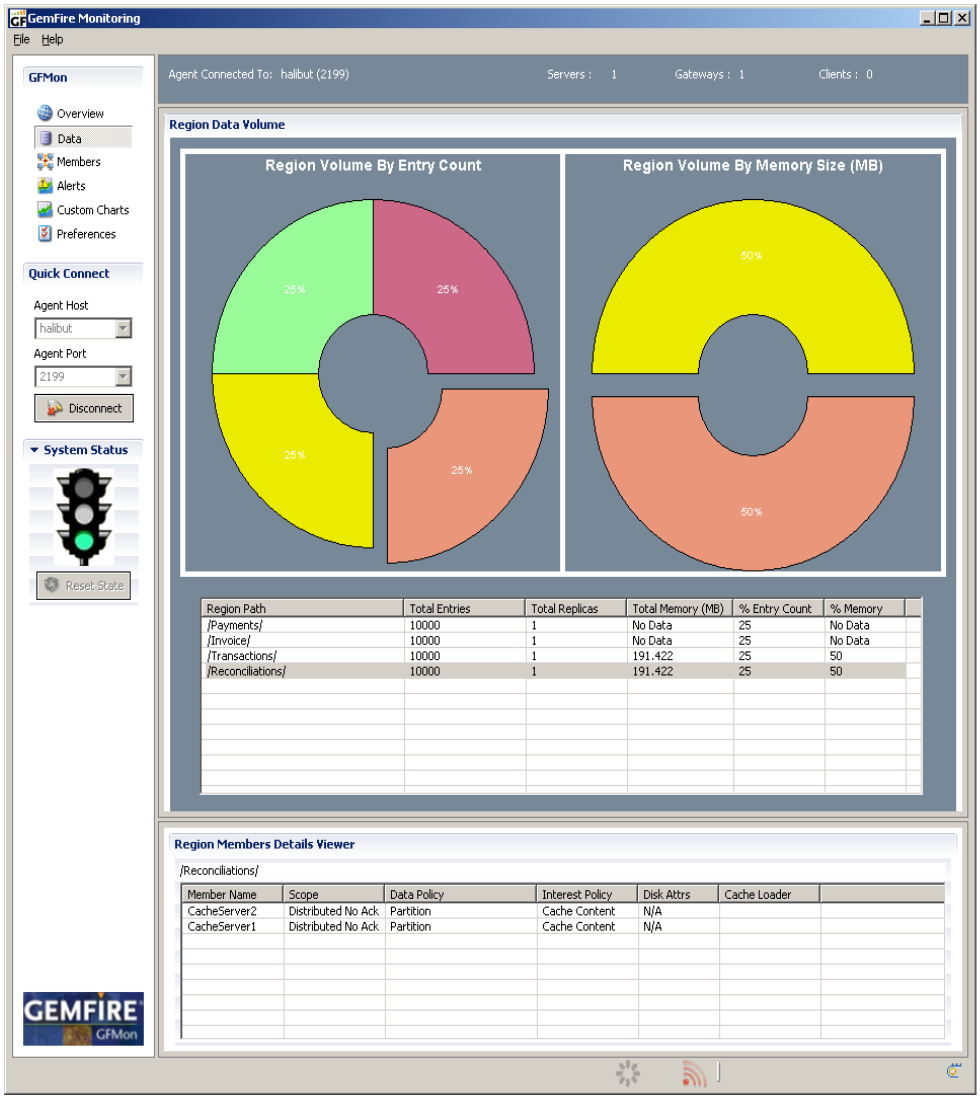


- Data Volume - This bar chart displays the entry count and entry size of the data versus the regions.
- Gets Puts Ops - This graph plots the get and put operations occurring per second over the period of time specified.
- Regions Configurations - This table lists the attributes of the regions over which the cache is distributed. Region attributes like name, data policy, disk attributes, and backup persistence are provided.
- Gateway Hub - If the member that you have selected is a Gateway, the **Gateway Hub** section is displayed. This section displays the listening port and the Gateway ID. The Find Points table displays the following:
  - ID-The ID of the Gateway end point
  - Host/Port- The host name and port ID of the Gateway end-point
  - Connected-Whether the gateway is connected or not
- All Graphs - Displays all the graphs available for each member. Click each chart thumbnail to open the chart. Click **New Chart** to add a new chart. For more information, see *Custom Charts Panel*.

Data Panel

The **Data** panel provides a bird's eye view of the GemFire Enterprise data regions using pie charts. The **Data** panel also provides details like scope, and data policy of each region across the members.

Figure 3.4 Data Region Panel


GemFire Monitoring
File Help
GFMon
Overview
Data
Members
Alerts
Custom Charts
Preferences
Quick Connect
Agent Host
halibut
Agent Port
2199
Disconnect
System Status
Reset State
Agent Connected To: halibut (2199)
Servers : 1
Gateways : 1
Clients : 0
Region Data Volume
Region Volume By Entry Count
Region Volume By Memory Size (MB)
25%
25%
25%
25%
50%
50%
Region Path | Total Entries | Total Replicas | Total Memory (MB) | % Entry Count | % Memory
/Payments/ | 10000 | 1 | No Data | 25 | No Data
/Invoice/ | 10000 | 1 | No Data | 25 | No Data
/Transactions/ | 10000 | 1 | 191.422 | 25 | 50
/Reconciliations/ | 10000 | 1 | 191.422 | 25 | 50
Region Members Details Viewer
/Reconciliations/
Member Name | Scope | Data Policy | Interest Policy | Disk Attrs | Cache Loader
CacheServer2 | Distributed No Ack | Partition | Cache Content | N/A
CacheServer1 | Distributed No Ack | Partition | Cache Content | N/A
GEMFIRE GFMon

- Regional Data Volume - This section depicts the logical schema view of the distributed system. The **Region Volume by Entry Count** pie chart shows the distribution of the regions by entry count. The **Region Volume by Memory Size** shows the distribution of the regions by the amount of memory used.

  *The **Region Volume by Memory Size** chart displays memory size only under if the cache overflows and data is stored in the PR Region, Disk Region, or Region with Eviction set using MemLRU. Otherwise, the memory statistics are not captured in GemFire Enterprise and are not displayed in GFMon.*

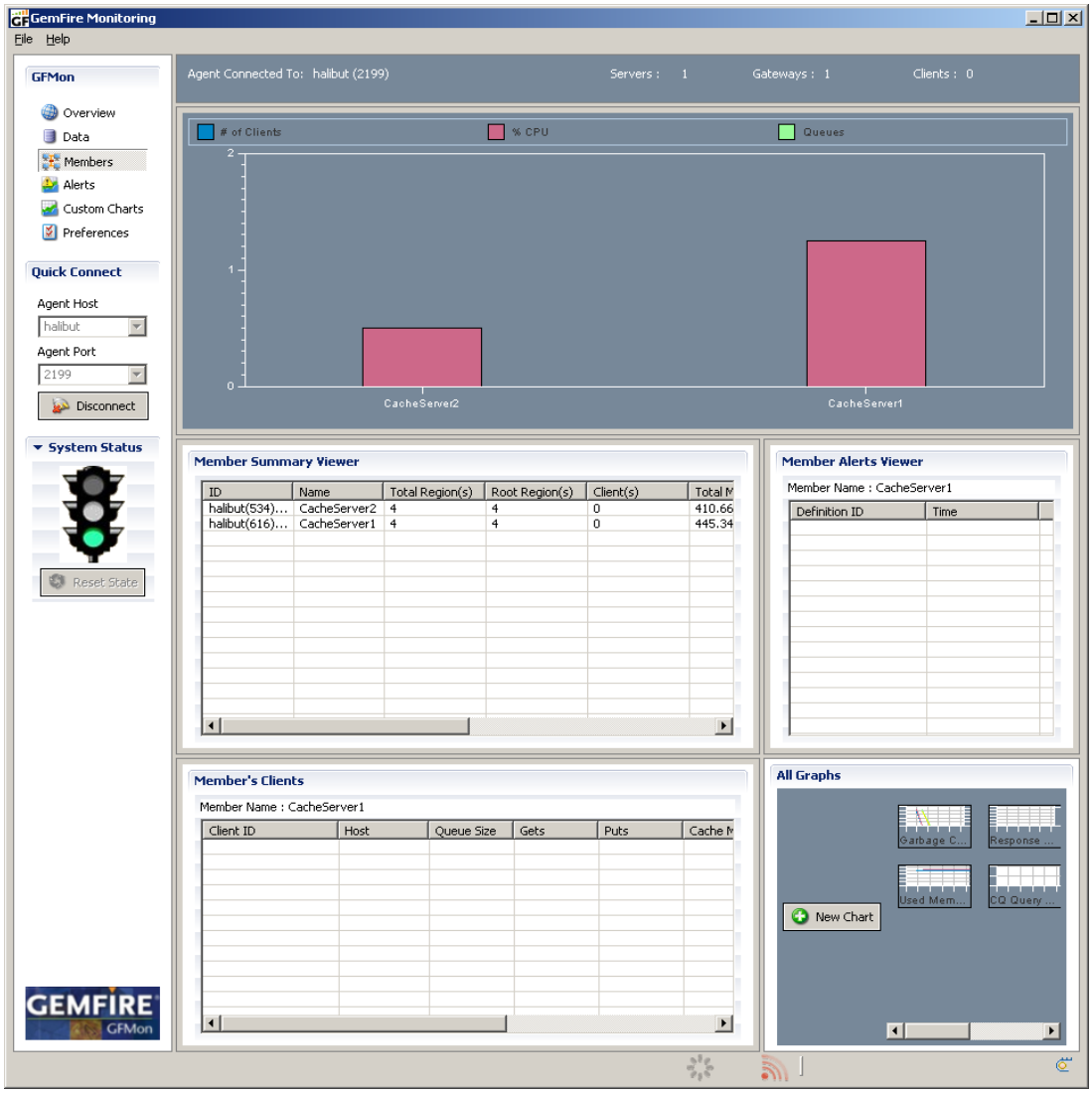
When you click on a pie in the chart, the details of your selection are highlighted in the table below the pie charts and related information is displayed in the **Region Members Details Viewer**.

- Region Members Details Viewer - This section provides a tabular view of the following region attributes:
  - Scope - For any region that is not partitioned, the region scope determines whether and how, region data is distributed between the local cache and the rest of the distributed system.
  - Data Policy - The data-policy attribute for each member determines which data is stored in the local cache.
  - Interest Policy - For each member, the interest-policy defines the entry operations that are delivered to the local cached region.
  - Disk Attributes - Disk attributes determine where, and how, region data is overflowed or persisted to a disk. You can define attributes templates inside the <cache> and <region> elements and assign IDs for later retrieval.
  - Cache Loader - A cache loader automatically loads data from an outside source, such as a database. In a distributed region that is not partitioned, one member may host the cache server for the entire distributed region. Loading into a partitioned region requires a cache loader in every partition.

Members Panel

The **Members** panel displays the characteristics of each member in the distributed system, alerts specific to the selected distributed system member, and allows you to view the various custom charts.

Figure 3.5 Members Panel



The animated bar chart in the **Members** panel displays the number of clients, CPU usage percentage, and the Queues for each member in the distributed system.

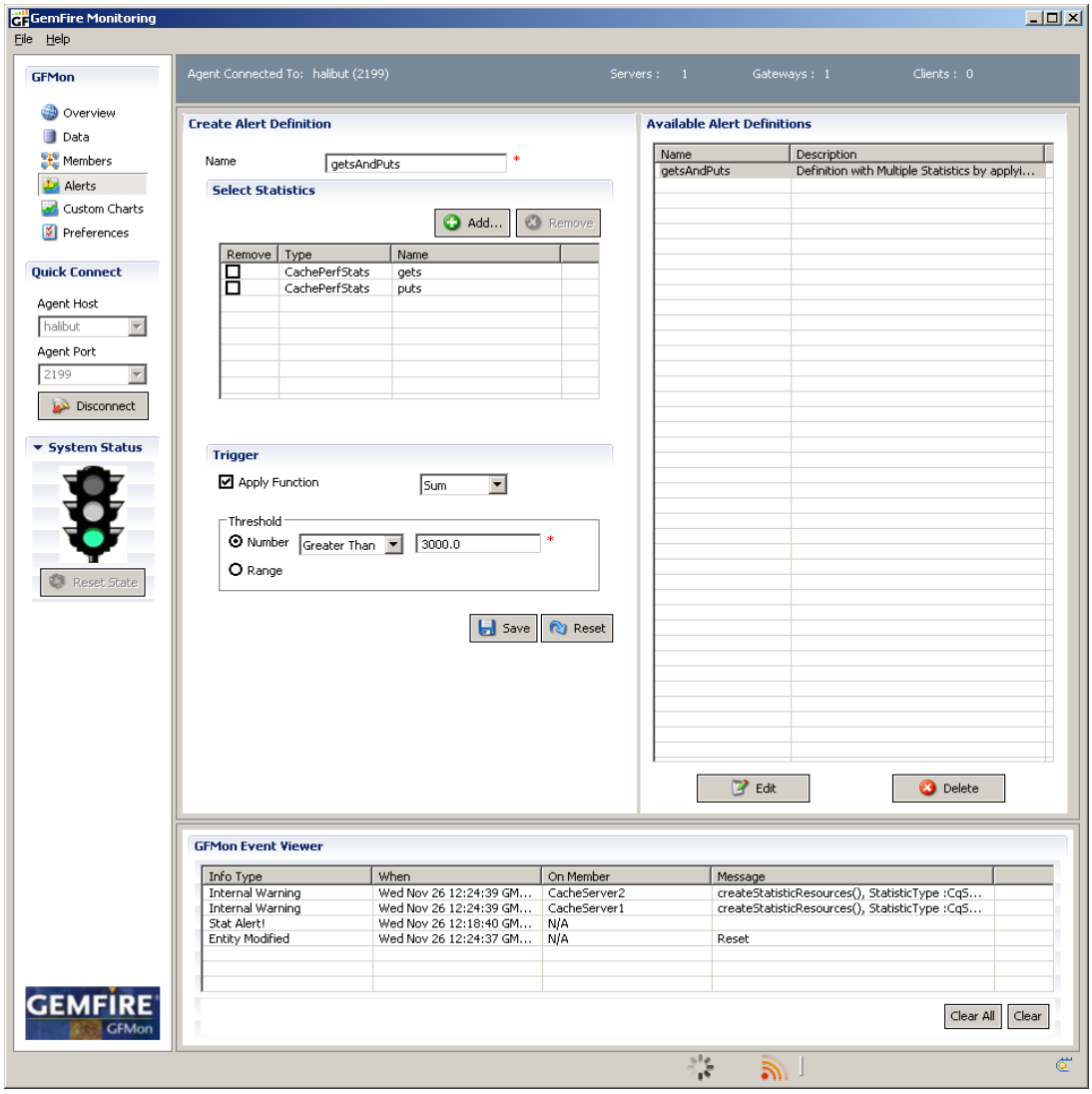
- Member Summary Viewer - This table lists the member attributes like ID, Name, Total Regions, Root Regions, Clients, Total memory (in MB), Gets/sec, Puts/sec, Threads, and Network Usage (KB). The member ID is a unique auto-generated value that identifies the member.
- Member Alerts Viewer - This table displays the Definition ID and Time for the alerts that you have configured for the member. This section displays the alerts that originated on the member since the last reset.
- Member's Clients - This table lists the details of the clients connected to each member. Select a member in the **Member Summary View** table to view the following attributes for each of the member's clients:
  - Client ID - a unique auto-generated ID identifies the client
  - Host - The name of the host the client is running on
  - Queue Size - The queue size for this client on the server that this client is connected to
  - Gets - number of gets/sec that the client is executing on the cache
  - Puts - number of puts/sec that the client is executing on the cache
  - Cache Misses - The number of times a get operation on the client cache resulted in the data being fetched from a server because it was not already present in the client cache
  - CPU Usage - CPU usage of the client
  - Threads - total number of threads in the client application
  - Cache Listener Invocations - the number of times the cache listener has been invoked.
- All Graphs - This section displays all the canned charts and the custom charts that you created in the **Custom Charts** panel. Click each icon to get a comparative real time view of the statistic being plotted. Click **New Chart** to add a new chart. For more information, see *Custom Charts Panel*.

Alerts Panel

You can create custom alerts in the **Alerts** panel. Custom alerts in GFMon allow you to receive a notification when a specified system statistic of an active member in a distributed system reaches a defined threshold. GemFire JMX Admin Agent alerts GFMon (and the RMI clients) when the customized alert reaches the specified threshold. For more information about receiving e-mail alerts, see *E-Mail Notification Properties* in the *GemFire Enterprise® System Administrator's Guide*.

For detailed description of system statistics, see the appendix *System Statistics* in the *GemFire Enterprise System Administrator's Guide*.

Figure 3.6 Alerts Panel



To create custom alerts in GFMon:

1. In the **Create Alert Definitions** section, enter a name for the alert.
2. In the **Select Statistics** section, click **Add...**. The **Choose Stats** window is displayed.

3. From the drop-down menu, choose the statistic type. The available statistics for the type you have chosen are displayed in the **Statistic Name** section.
4. Select the **Statistic Name(s)** and click the right-arrow to move them to the **Statistics Type (Statistic name)** section. If you want to deselect any values, select the values in the **Statistics Type (Statistic name)** section and click the left-arrow button.
5. Click **Select**. The statistics that you have selected are displayed in the **Select Statistics** table. You can add more values to the table by repeating the above steps. To remove a value, select the check-box in the **Remove** column and click **Remove**.
6. If you have selected more than one statistic value, you can apply a function to the statistics to create alerts. In the **Trigger** section, select the check-box beside **Apply Function**. The drop-down list is populated with the available functions.
7. In the **Trigger** section, select the function that you want to use and specify either the **Number** (for which the statistic value must be greater than or less than) or the **Range** (within which the statistic value must fall).

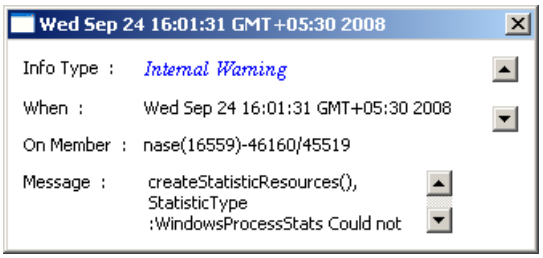
   You cannot apply a function if you have selected only one statistic value.

8. Click **Save** to save the alert definition. The alert name is now displayed in the **Available Alert Definitions** section.
9. To modify an alert definition, select the alert and click **Edit**. You can change the alert definition by editing the desired parameter. To delete an alert definition, select the alert and click **Delete**.

All alerts are displayed in the **Alerts!** section of the **Overview** panel.

The **GFMon Event Viewer** table displays warnings that are logged in the `GFMon.log` file. You can view the internal warnings and node information in the **GFMon Event Viewer** and see the log file for details. Double-click any item in the table to get more information about the event. The **Info Type** scroll-bar allows you to scroll through all the events in the table. Use the **Message** scroll-bar to scroll through the complete message.

Figure 3.7 Details for Events in the GFMon Event Viewer



By default the **GFMon Event Viewer** displays the *last 100* events only. Go to the **General** tab in the **Preferences** panel to configure preferences:

- To change the number of events viewable in the **GFMon Event Viewer**, change the **Maximum number of rows viewable in a table** preference.
- To change the logging level of the `GFmon.log` file, change the **Logging Level** preference.

The following events are logged in the event viewer.

Table 3.1 Event types Logged in the GFMon Event Viewer

Event Type	Purpose
Node Joined	When a node joins the system
Node Crashed	When a node crashes
Node Left	When a node leaves the system
Alert!	When a system alert occurs
Internal Error	When a GFMon error occurs
Internal Warning	When a GFMon warning occurs
Reconnecting to the Agent!	When GFMon attempts to reconnect to JMX agent. This happens when JMX agent shuts down unexpectedly
Shutdown of the Agent!	When the attempt to reconnect to JMX agent fails

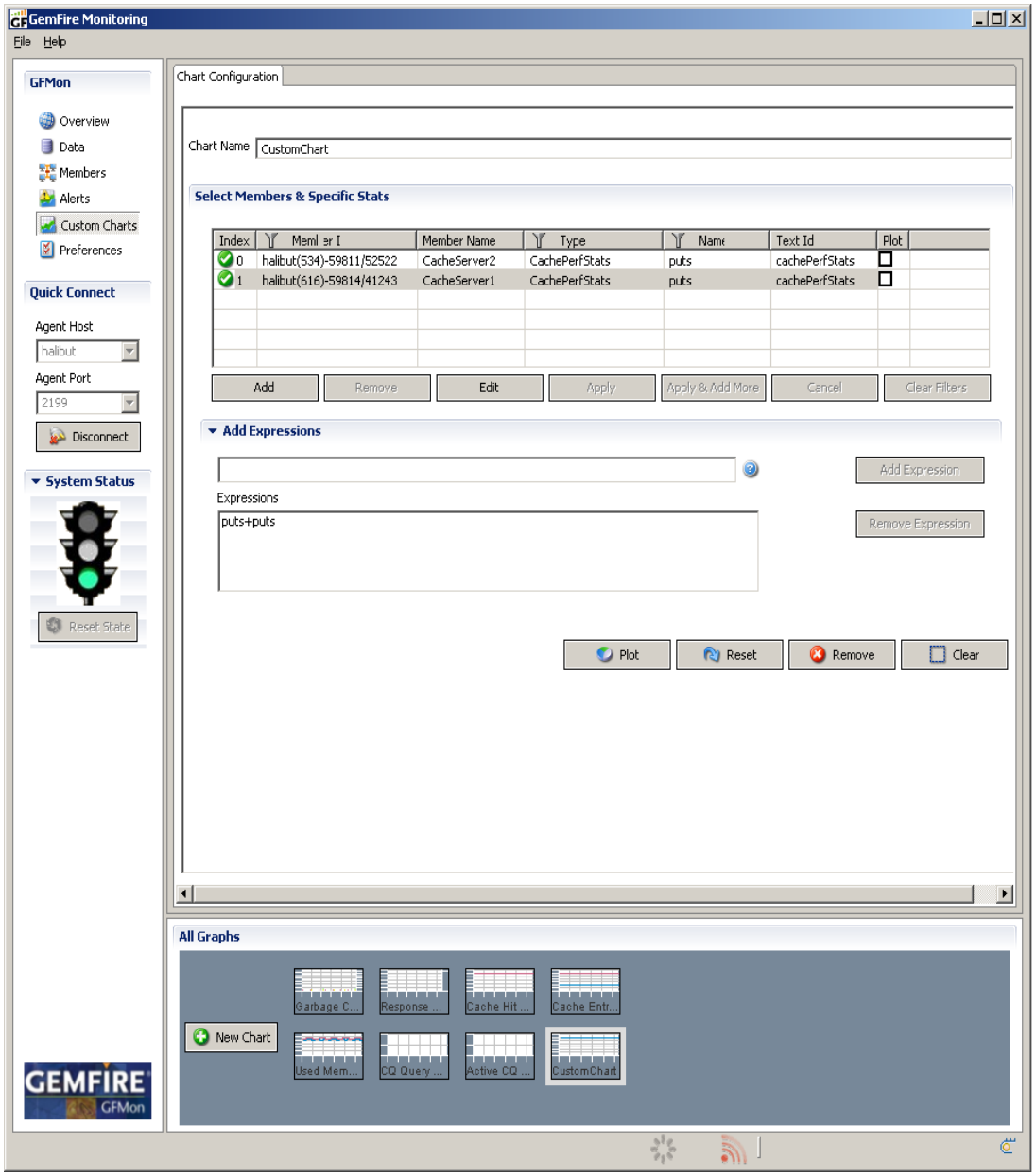
Use the **Clear** and **Clear All** buttons at the bottom-right corner of the **GFMon Event Viewer** panel to clear the selected event or to clear all events respectively.

Custom Charts Panel

GFMon provides canned charts that graphically display the state of the distributed system. The canned charts are displayed in the **All Graphs** section. Each graph is plotted with the selected parameter on the Y-axis and time on the X-axis. Click the icon to open each chart. The top of the chart displays the legend of each member. Select a member name to highlight the line graph of the member data and vice versa.

You can also create additional charts in the **Custom Charts** panel.

Figure 3.8 Custom Charts — Chart Configuration Tab



To create a chart:

1. In the **All Graphs** section, click **New Chart**.
2. Enter a Chart Name.
3. In the **Select Members & Specific Stats** section, click **Add**. From the **Member ID** drop-down menu, select a member. The **Member Name** is displayed.
4. Select the statistic **Type**, **Name**, and **Text ID**. Click **Apply** to apply the statistic to the graph. Click **Apply & Add More** to add another statistic. Ensure that you have applied all statistics before you proceed.
5. If you want to add expressions, add the statistic values to the **Add expressions** field and click **Add Expression**. For information about the expression syntax, click the **?** symbol.
6. Click **Plot** to plot the graph that you created.
7. To reset the chart options, click **Reset**. To delete a chart, select the chart and click **Delete**.

When you click **New Chart**, the **Custom Charts** panel is cleared of any existing chart configuration enabling you to create a new chart. Also, when you click **New Chart**, GFMon hides all the open chart windows so that the **Custom Chart** panel does not remain hidden behind the active windows. After plotting the new chart by clicking **Plot**, you can click on any of the Custom Charts to re-open them.

> *When you **edit** the name of an existing chart, a new copy of the current chart with the new name is created.*

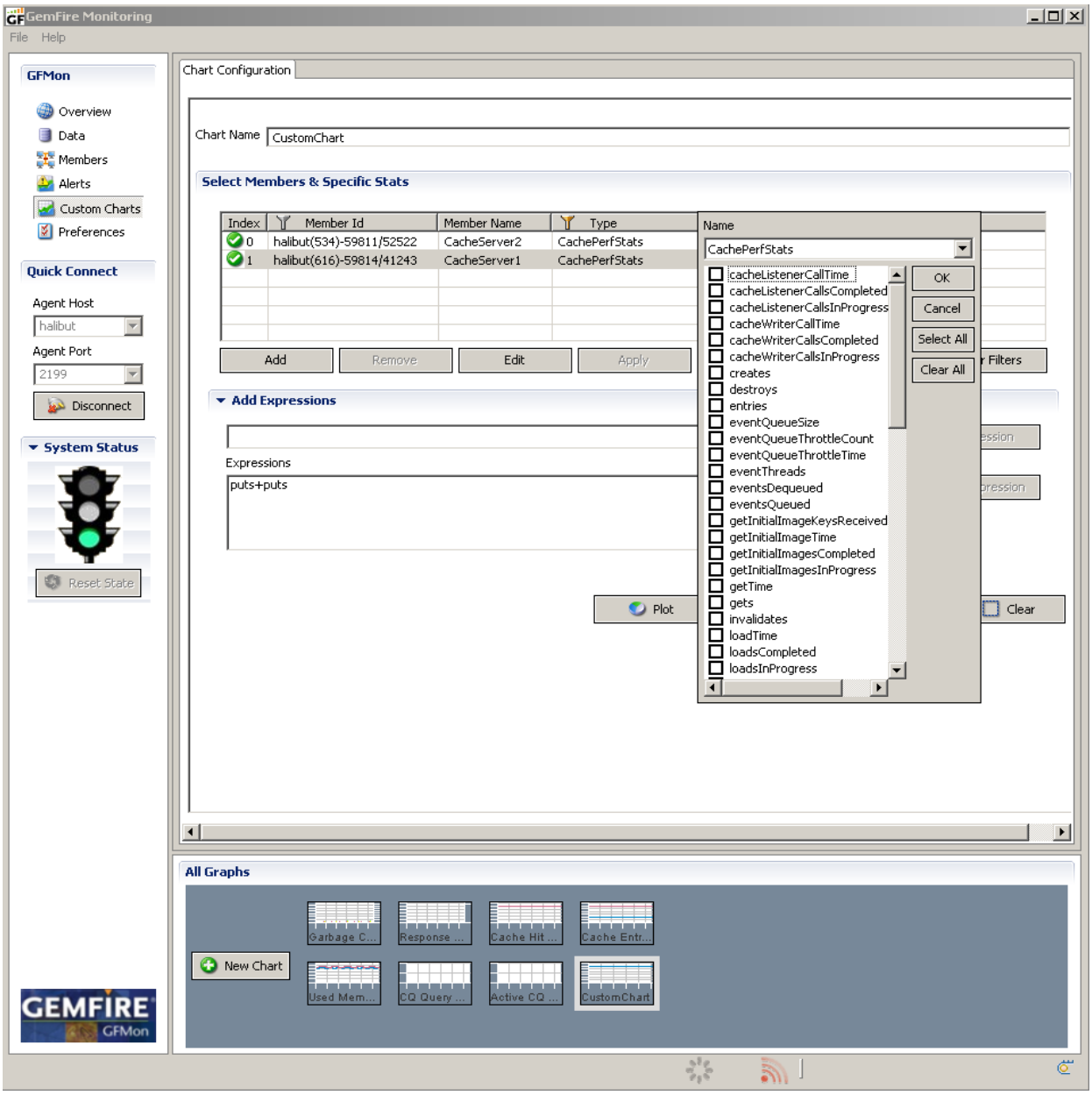
For detailed description of system statistics, see the appendix *System Statistics* in the *GemFire Enterprise System Administrator's Guide*.

Filters can be applied to the **Member ID**, **Statistic Type,** and **Name** columns.

Filters allow you to quickly create charts in which you are interested only in a subset of the members and statistics. The columns that offer the ability to filter their data have a funnel icon in their headers. Filters can be selected at any time during the creation of a chart. The filters get applied only to the rows created after you define the filter.

For example, you can set a filter for a statistic type and name to compare one statistic across all members.

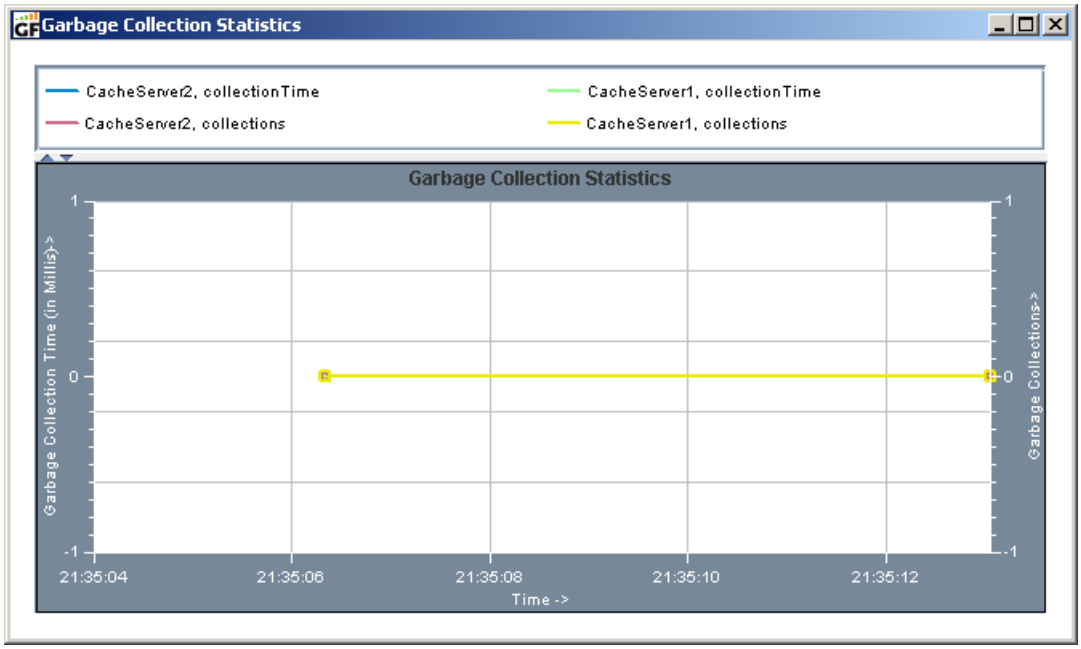
Figure 3.9 Using Filters in the Custom Charts Panel



The following canned charts are available in GFMon. You can view these charts when you connect to a GemFire Enterprise system. These charts plot line graphs of the respective statistic versus the duration of time specified through the **X Scale Time range for Time Charts (in minutes)** preference, in the **Preferences - General** panel:

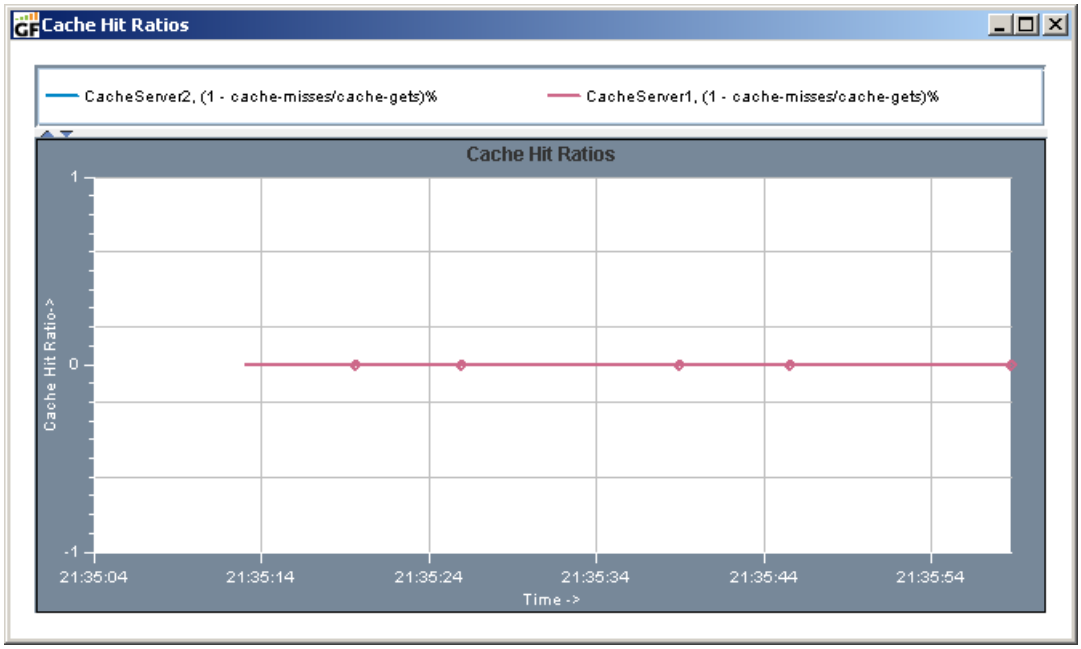
- Garbage Collection Statistics - This chart displays the number of GC collections and the time spent in the collection for each member's VM.

Figure 3.10 Garbage Collection Statistics Chart



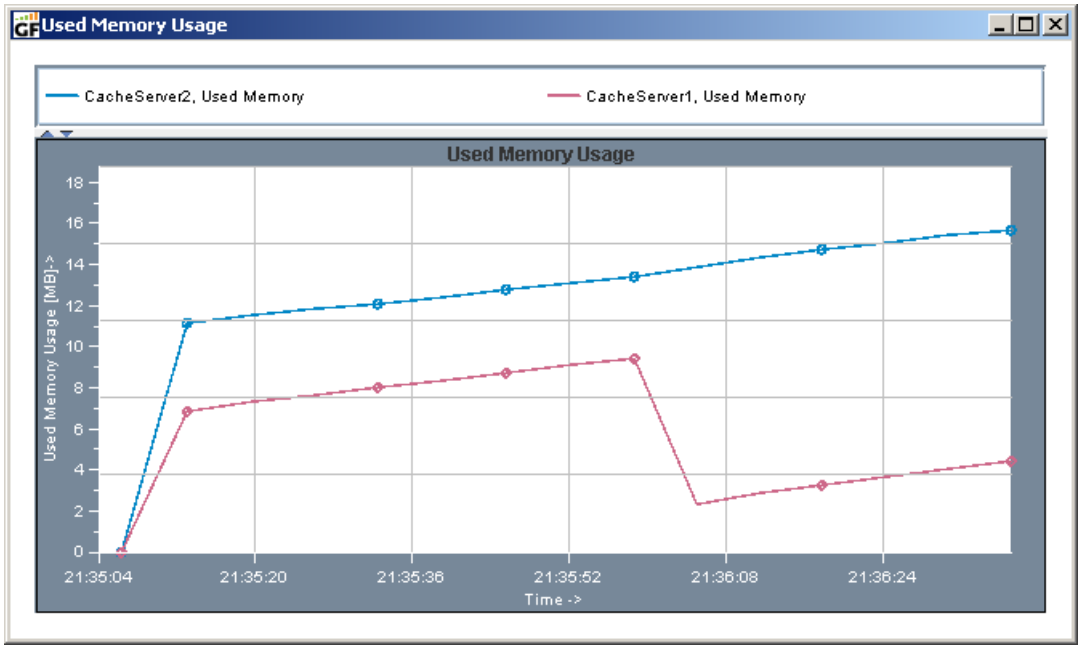
- Cache Hit Ratios - This chart displays the number of times a `get` on a cache did not result in the value being found in local memory and resulted in a netload from the cache server.

Figure 3.11 Cache Hit Ratios Chart



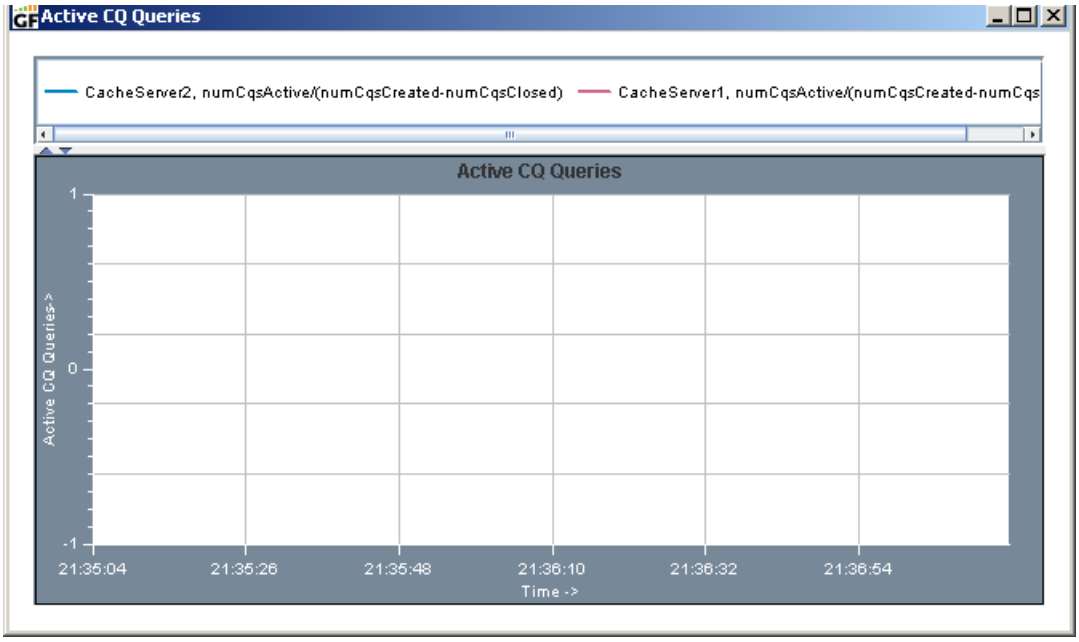
- Used Memory Usage - This chart displays the amount heap memory; in megabytes; consumed by each active member in the distributed system over time.

Figure 3.12 Used Memory Usage Chart



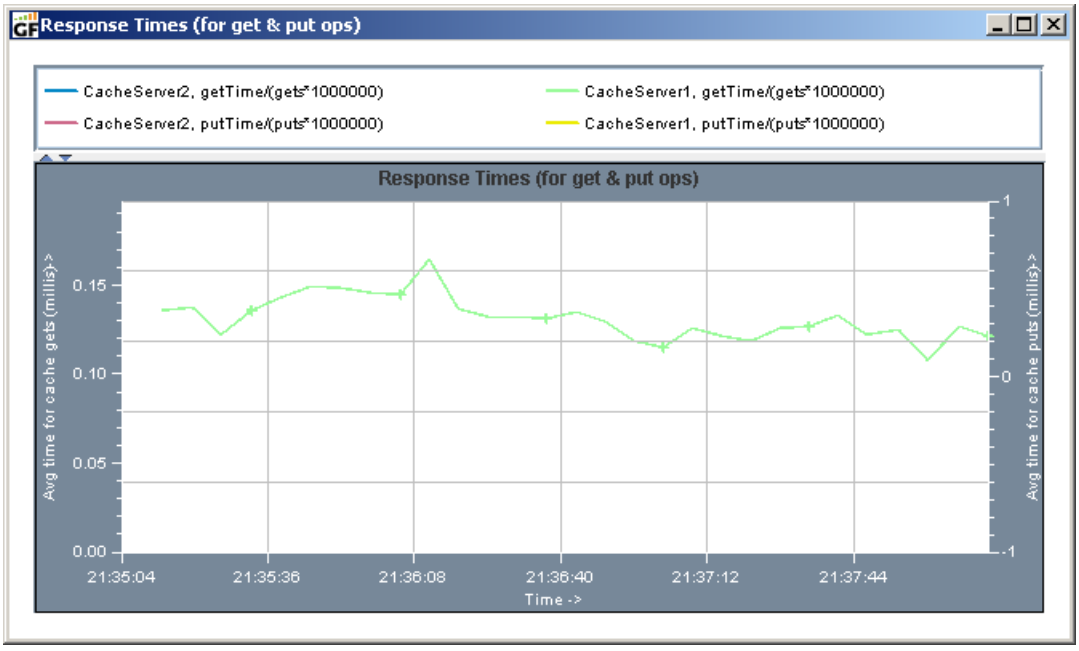
- Active CQ Queries - This chart displays the number of active CQs as a percentage of total CQs.

Figure 3.13 Active CQ Queries Chart



- Response Times (for get & put operations) - This chart plots the average time spent in `get` and `put` operations for each member in the distributed system.

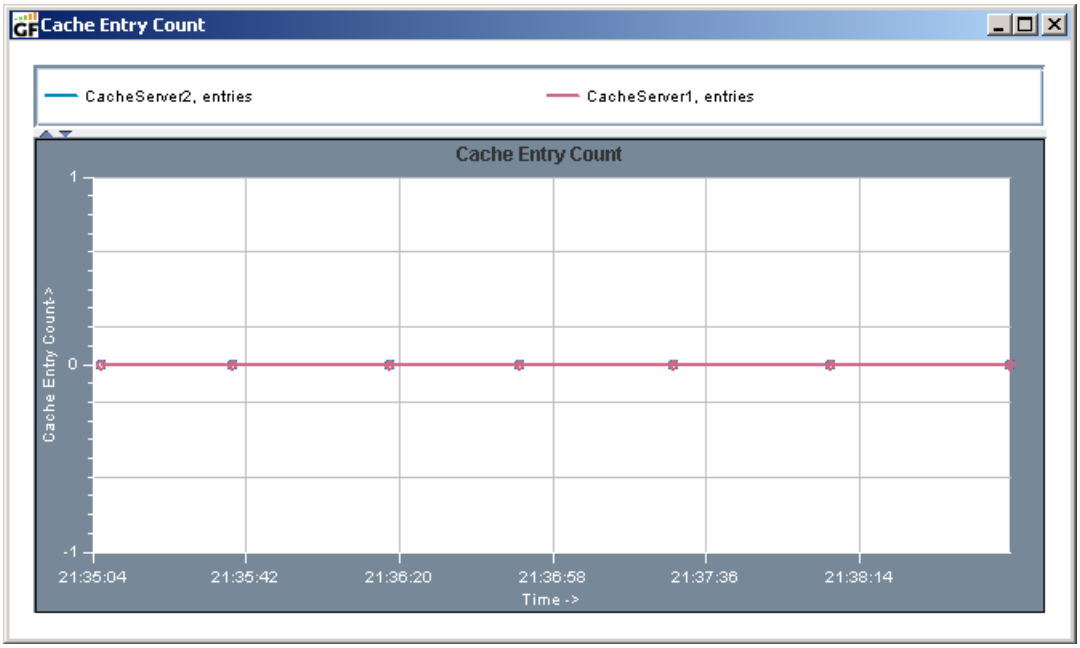
Figure 3.14 Response Times Chart



All charts are computed incrementally. If the time statistics are not enabled in the member, the Response Times chart does show any data.

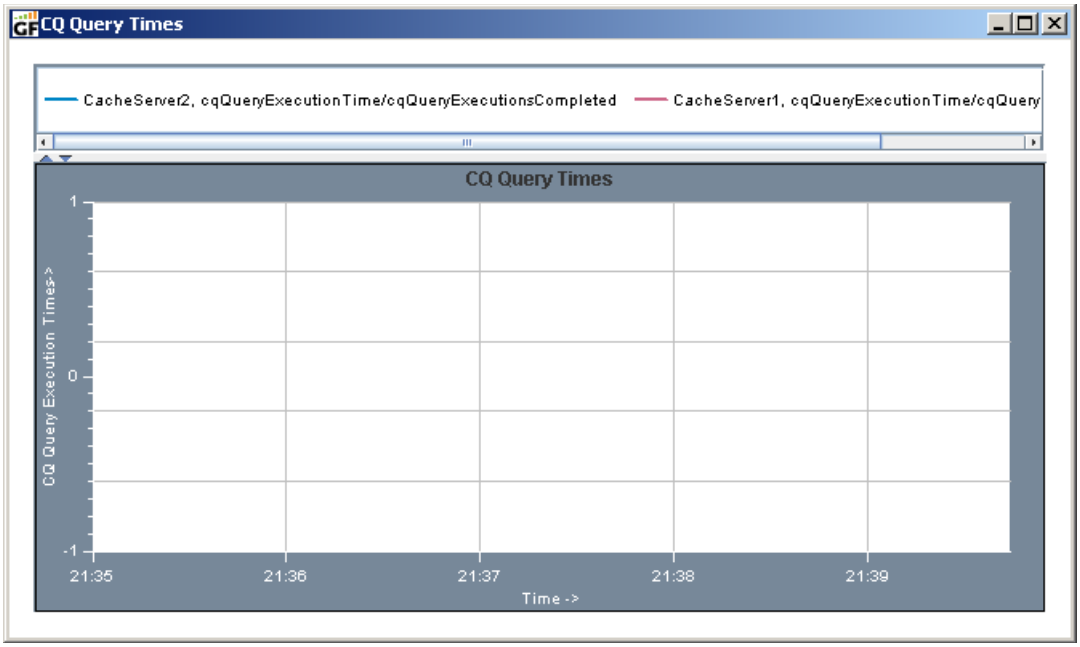
- Cache Entry Count - This chart displays the total number of entries in all the non-partitioned data regions on a cache server over time.

Figure 3.15 Cache Entry Count Chart



- CQ Query Times - This chart displays the average time; in milliseconds; taken to execute a CQ.

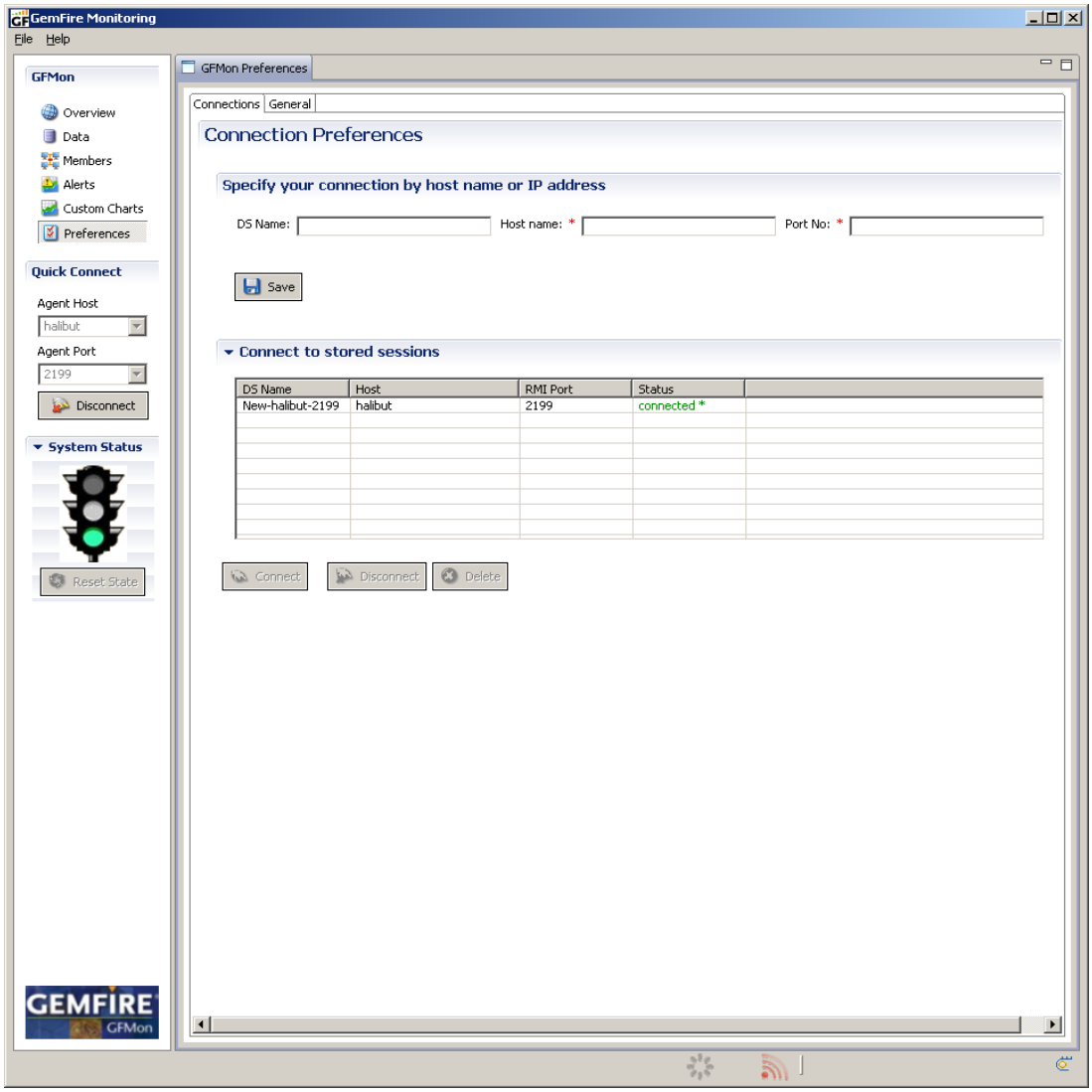
Figure 3.16 CQ Query Times Chart



Preferences Panel

The **Preferences** panel allows you to set connection and general preferences and save them for future use. On Windows platforms, these preferences are saved in the system registry in the logged in the `HKEY_CURRENT_USER\Software\JavaSoft\Prefs\GemFire Monitor 2.0\<Logged-In-User-Name>` directory. On Linux platforms, these preferences are saved in the `/home/<Logged-In-User-Name>/.java/.userPrefs/home/{Logged-In-User-Name}/prefs.xml` file.

Figure 3.17 Preferences — Connections Tab



You can use the **Connections** tab to create connection preferences and store them for future use. To create and use connection preferences:

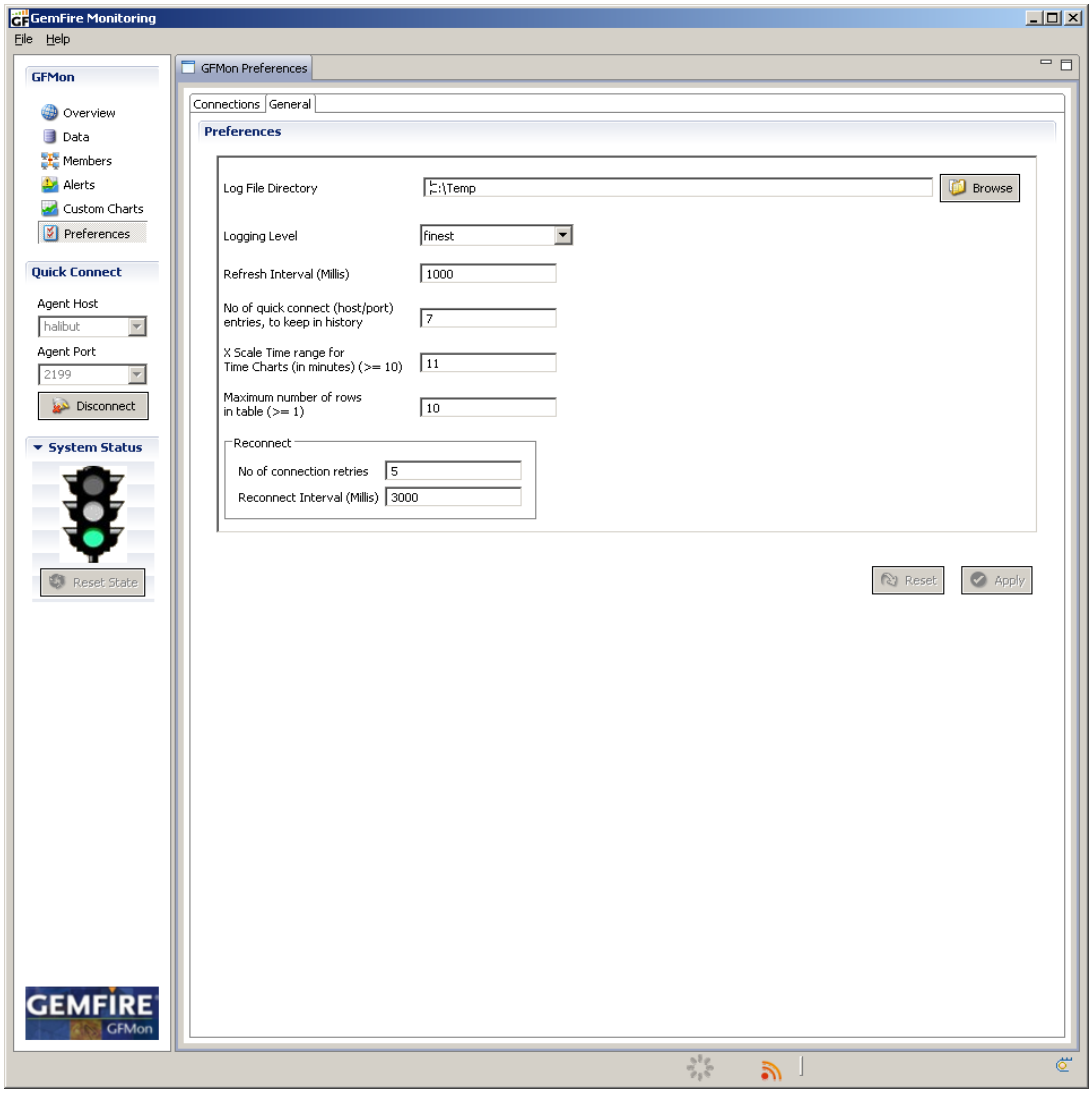
1. In the **Specify your connection by host name or IP address** section, enter system name, host name, and port number.
2. Click **Save**. The information that you entered is populated in the **Connect to stored sessions** section.

   *If you connect to a new agent, it is automatically populated in the **Connect to stored sessions** section. The un-saved connections are denoted by an appended ***** (asterisk) in the **Status** column.*

3. Highlight the row of the system name and click **Connect** or **Disconnect** as required.

   At any given time, you can connect to one distributed system only.

Figure 3.18 Preferences — General Tab



You can use **General** tab to set various preferences of your GFMon tool. You can configure the following preferences:

- Log file Directory - Use the **Browse** option to specify the directory where you want the GFMon log files to be stored. Ensure that the GFMon application has read and write access to the log files.
- Logging Level - Select the logging level from the available options in the drop-down menu. The default value is **info**. The valid logging levels are **all**, **finest**, **finer**, **fine**, **config**, **info**, **warning**, **error**, **severe**, and **none**. The logging level **all** provides information of all severity levels, while the logging level **severe** provides the information of the highest severity level.

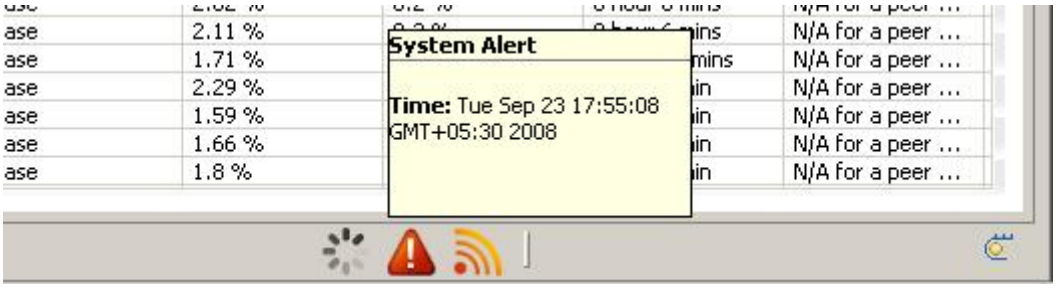
  Setting log-level to one of the ordered levels causes all messages of that level and greater severity to be printed. Lowering the log-level reduces system resource consumption while still providing some logging information for failure analysis.
- Refresh Interval (in milliseconds) - Specify the refresh interval for the data displayed in the **Data** panel.
- Number of quick connect (host/port) entries to keep in history - Define the number of entries the Agent host and Agent port fields must retain in the **Quick Connect** section.
- X scale time range for Time Charts (in minutes) - Define the time for which the charts must be plotted in the **Custom Charts** panel.
- Reconnect - Specify the number of times and time interval GFMon must attempt to reconnect (in the **Number of retries** and **Reconnect Interval in milliseconds** fields) if the connection is lost.
- Maximum number of rows in a table - The maximum number of rows that are kept in a table. For example, you can configure the number of entries in the **Alerts!** table in the **Overview** panel and the **GFMon Event Viewer** table in the **Alerts** panel.

After selecting the option or changing the value that you require, click **Reset** or **Apply** as applicable. The changes that you apply reflect immediately, except changes to **Log File Directory** and **Refresh Interval** which reflect only when you launch GFMon for the next time.

Status Bar

The status bar at the bottom of your GFMon window provides various alerts.

Figure 3.19 Status Bar



The following icons are displayed at various times indicate the following:

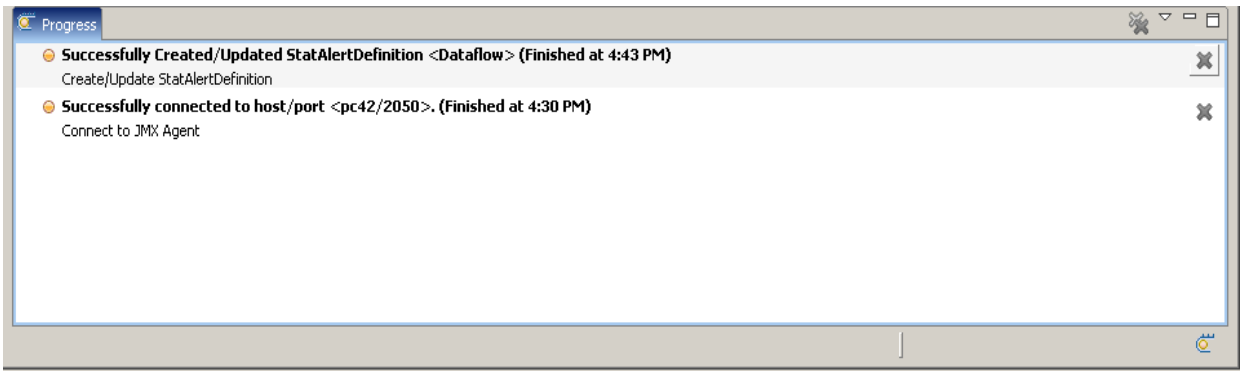
- This Icon appears when a JMX connection is established.

or - These icons are displayed when any alerts occur.

- This icon is displayed when any background jobs are in progress.

- This icon is displayed after you connect to the distributed system and contains information about the agent connection, agent disconnection, and the alerts that you save. Click this icon to expand or collapse the **Progress** view shown in the illustration below.

Figure 3.20 The Progress View


Progress
Successfully Created/Updated StatAlertDefinition <Dataflow> (Finished at 4:43 PM)
Create/Update StatAlertDefinition
Successfully connected to host/port <pc42/2050>. (Finished at 4:30 PM)
Connect to JMX Agent

Disconnecting From the Agent in GFMon

When you disconnect from the agent in GFMon or if the JMX agent shuts down, the data in the panels is not cleared. When you connect to any distributed system again, the panels are populated with the new data.

3.2 Error Notifications Displayed by GFMon

During failure situations described below, GFMon displays a pop-up window reporting a fatal error. The error pop-up window is shown when the following conditions occur:

- The JMX agent that GFMon is connected to shuts down. A pop-up window titled **Fatal Error** is displayed with the following message:

  ```
  JMX Agent has shutdown
  ```

- GFMon is started and running with a `gemfire.jar` file which is incompatible with the version of the `gemfire.jar` file of the running JMX Agent. In this condition, a pop-up window titled **Fatal Error** is displayed:
  - For example, if you run GFMon with the `gemfire.jar` file of GFMon version 5.7 to monitor a GFMon 5.7.1 or GFMon 5.8 distributed system, the message is: `GFMon could not connect to the GemFire Distributed System. This version of GFMon is compatible with GemFire Enterprise 5.7. GFmon encountered a Class Mismatch. Please shutdown GFMon and restart with a compatible GemFire version`
- GFMon encounters an error during run-time while retrieving data from the admin agent. A pop-up window titled **Fatal Error** is displayed with a message depending on the error that has occurred.

To fix the issue, close the pop-up window and perform the following procedure:

1. Turn off GFMon.
2. Reset the environment variable `GEMFIRE` to point to the GFMon product location that the admin agent is running from.
3. Restart GFMon.

Glossary

ACK wait threshold A time-to-wait for message acknowledgement between system members.

administrative event See event.

API **A**pplication **P**rogramming **I**nterface. GemFire provides APIs to cached data for Java applications.

application program A program designed to perform a specific function directly for the user or, in some cases, for another application program. GemFire applications use the GemFire application programming interfaces (APIs) to modify cached data.

attribute Querying: A named member of a data object. The public fields and methods of an object may be accessed as attributes in the context of a query.

Region: See region attributes.

attribute path A sequence of attributes separated by a dot (.), applied to objects where the value of each attribute is used to apply the next attribute.

blocking A behavior associated with synchronization functions. Blocking behavior is exhibited as waiting for a signal to proceed, regardless of how long it takes. See also timeout.

cache A data cache created by an application or cache server. This is the point of access for Java applications for all caching features, and the only view of the cache that is available to the application. Cache creation requires a connection to the distributed system. See also local and remote.

cache-local Residing or occurring in the local cache.

cache.xml An XML file that declares the initial configuration of a cache. This file is used to customize the behavior of the GemFire cache server process and can be used by any Java application. Applications can also configure the cache through the GemFire Java APIs.

cache event See event.

cache listener User-implemented plug-in for receiving and handling region entry events. A region's cache listener is called after an entry in the local cache is modified. See also cache writer.

cache loader User-implemented plug-in for loading data into a region. A region's cache loader is used to load data that is requested of the region but is not available in the distributed system. For a distributed region, the loader that is used can be in a different cache from the one where the data-request operation originated. See also netSearch and netLoad.

cache server A long-lived, configurable GemFire distributed system member process.

cache transaction A native GemFire transaction, managed by GemFire and not by JTA. This type of transaction operates only on data available from the GemFire cache in the local member. See also JTA and global transaction.

cache writer User-implemented plug-in intended for synchronizing the cache with an outside data source. A region's cache writer is a synchronous listener to cache data events. The cache writer has the ability to abort a data modification. See also cache listener and netWrite.

collection Used in the context of a query for a group of distinct objects of homogeneous type, referred to as elements. Valid collections include the `java.util.Collection` as well as `Set`, `Map`, `List`, and arrays. The elements in a collection can be iterated over. Iteration over a `Map` traverses its entries as instances of `Map.Entry`. A region can also be treated as a collection of its values. See also QRegion.

commit A transactional operation that merges a transaction's result into the cache. Changes are made in an "all or none" fashion. Other changes from outside the current transaction are kept separate from those being committed.

concurrency-level Region attribute that specifies an estimate of the number of threads ever expected to concurrently modify values in the region. The actual concurrency may vary; this value is used to optimize the allocation of system resources.

conflation Combining entries in a message queue for better performance. When an entry update is added to the queue, if the last operation queued for that key is also an update operation, the previously enqueued update is removed, leaving only the latest update to be sent to the consumer. Only entry update messages in a region with `distributed-no-ack` scope are conflated. Region operations and entry operations other than updates are not conflated.

connection The connection used by a Java application to access a GemFire Enterprise system. A Java application can connect to a GemFire Enterprise system by calling the `com.gemstone.gemfire.distributed.DistributedSystem.connect` method with the appropriate parameter settings. A Java application must connect to a distributed system to gain access to the GemFire Enterprise functionalities.

coordinator The member of the distributed system that sends out membership views. This is typically the locator in GemFire.

data-policy Region attribute used to determine what events the region receives from remote caches, whether data is stored in the local cache, and whether the data is persisted to disk. For disk persistence, writes are performed according to the region attribute settings, disk-dirs and disk-write-attributes.

deadlock A situation in which two or more processes are waiting indefinitely for events that will never occur.

destroy Distributed**:** To remove a cached object across the distributed cache.

Local**:** To remove a cached object from the local cache only.

disk-dirs Region attribute used to specify where to store region data on disk. Region entry data is stored on disk when the data-policy region attribute set to persistent-replicate and when an eviction controller is used with an overflow eviction action specified.

disk region A persistent region.

disk-write-attributes Region attribute used to specify how to write data to disk. This specifies whether writes are synchronous or asynchronous and, if asynchronous, it also specifies the frequency of writes. This is used in conjunction with the disk-dirs attribute when persistence or overflow are specified.

distributed cache A collection of caches spread across multiple machines and multiple locations that functions as a single cache for the individual applications.

distributed system One or more GemFire system members that have been configured to communicate with each other, forming a single, logical system. Also used for the Java object that is instantiated to create the connection between the distributed system members.

distributed-ack scope Data distribution setting that causes synchronous distribution operations, which wait for acknowledgement from other caches before continuing. Operations from multiple caches can arrive out of order. This scope is slower but more reliable than `distributed-no-ack`.

distributed-no-ack-scope Data distribution setting that causes asynchronous distribution operations, which return without waiting for a response from other caches. This scope produces the best performance, but is prone to race conditions.

entry A data object in a region. A region entry consists of a key and a value. The value is either null (invalid) or a Java object. A region entry knows what region it is in. An entry can have an application-defined user attribute. See also region data, entry key, and entry value.

event An action recognized by the GemFire system members, which can respond by executing callback methods. The GemFire API produces two types of events: cache events for detail-level management of applications with data caches and administrative events for higher-level management of the distributed system and its components. An operation can produce administrative events, cache events, or both.

entry key The unique identifier for an entry in a region.

entry value The data contained in an entry.

eviction-attributes Region attribute that causes the cache to limit the size of the region by removing old entries to make space for new ones.

expiration A cached object expires when its time-to-live or idle timeout counters are exhausted. A region has one set of expiration attributes for itself and one set for all of its entries.

expiration action The action to be taken when a cached object expires. The expiration action specifies whether the object is to be invalidated or destroyed and whether the action is to be performed only in the local cache or throughout the distributed system. A destroyed object is completely removed from the cache. A region is invalidated by invalidating all entries contained in the region, including those in subregions. An entry is invalidated by having its value marked as invalid. `Region.getEntry.getValue` returns `null` for an invalid entry.

In GemFire, expiration attributes are set at the region level for the region and at the entry level for entries. See also idle timeout and time-to-live.

factory method An interface for creating an object which at creation time can let its subclasses decide which class to instantiate. The factory method helps instantiate the appropriate subclass by creating the correct object from a group of related classes.

forced disconnect Forcible removal of a member from membership without the member's consent.

gemfire Command-line utility that allows you to start, stop, and otherwise manage GemFire locators from an operating system command prompt. The utility also provides license information and log management and merging.

global scope Data distribution setting that provides locking across the distributed system for load, create, put, invalidate, and destroy operations on the region, its subregions, and entries. This scope is the slowest, but it guarantees consistency across the distributed system.

global transaction A JTA-controlled transaction in which multiple resources, such as the GemFire cache and a JDBC database connection, participate. JTA coordinates the completion of the transaction with each of the transaction's resources. See also JTA and cache transaction.

HTTP World Wide Web's **H**yper**t**ext **T**ransfer **P**rotocol. A standard protocol used to request and transmit information over the Internet or other computer network.

idle timeout The amount of time a region or region entry may remain in the cache without being accessed before being expired. Access to an entry includes any get operation and any operation that resets the entry's time-to-live counter. Region access includes any operation that resets an entry or subregion idle timeout and any operation that resets the region's time-to-live.

Idle timeout attributes are set at the region level for the region and at the entry level for entries. See also time-to-live and expiration action.

initial capacity Region attribute. The initial capacity of the map used for storing region entries.

invalid The state of an object when the cache holding it does not have the current value of the object.

invalidate Distributed**:** To mark an object as being invalid across the distributed cache.

Local**:** To mark an object as being invalid in the local cache only.

JDBC **J**ava **D**ata**B**ase **C**onnectivity. A programming interface that lets Java applications access a database via the SQL language.

JMX **J**ava **M**anagement e**X**tensions. A set of specifications for dynamic application and network management in the J2EE development and application environment.

JNDI **J**ava **N**aming and **D**irectory **I**nterface. An interface to naming and directory services for Java applications. Applications can use JNDI to locate data sources, such as databases to use in global transactions. GemFire allows its JNDI to be configured in a `cache.xml` configuration file.

JTA **J**ava **T**ransaction **A**PI. The local Java interfaces between a transaction manager (JTS) and the parties involved in a global transaction. GemFire can be a member of a JTA global transaction. See also global transaction.

key constraint Enforcing a specific entry key type. The `key-constraint` region attribute, when set, constrains the entries in the region to keys of the specified object type.

listener An event handler. The listener registers its interest in one or more events, such as region entry updates, and is notified when the events occur.

load factor Region attribute. The load factor of the map used for storing entries.

local local cache: The part of the distributed cache that is resident in the current member's cache. This term is used to differentiate the cache where a specific operation is being performed from other caches in the distributed system. See also remote.

region with local scope: A region whose scope is not distributed.

local scope Data distribution setting that keeps data private and visible only to threads running within the local member. A region with local scope is completely contained in the local cache.

LRU **L**east **R**ecently **U**sed. Used to refer to region entry or entries most eligible for eviction due to lack of interest by client applications. GemFire offers eviction controllers that use the LRU status of a region's entries to determine which to evict to free up space. Possible eviction actions are local destroy and overflow.

message queue A first-in, first-out data structure in a GemFire Enterprise system member that stores messages for distribution in the same order that the original operations happened in the local member. Each thread has its own queue. Depending on the kind of queue, the messages could be going between two members of a distributed system, a client and server, or two members in different distributed systems. See also conflation.

mirroring See replicate.

multicast A form of UDP communications where a datagram is sent to multiple processes in one network operation.

netLoad The method used by GemFire Enterprise to load an entry value into a distributed region. The `netLoad` operation invokes all remote cache loaders defined for the region until either the entry value is successfully loaded or all loaders have been tried.

netSearch The method used by GemFire Enterprise to search remote caches for a data entry that is not found in the member's local cache region. This method operates only on distributed regions.

netWrite The method used by GemFire Enterprise to invoke a cache writer for region and region entry events. This method operates only on distributed regions. For each event, if any cache writer is defined for the region, the `netWrite` operation invokes exactly one of them.

network partitioning A situation that arises from a communications partition that causes processes to become unaware of one another.

notify all This a server configuration option that applies to installations where clients have a pool with queueing enabled. This causes the server to send notifications for all entry updates. The notifications are all invalidations. They do not include the new entry values. Compare to notify by subscription.

notify by subscription This a server configuration option that applies to installations where clients have a pool with queueing enabled. This causes the server to send notification for any entry update only to those clients that have registered interest in the entry. The notification includes the entry value. Compare to notify all.

OQL — **O**bject **Q**uery **L**anguage, SQL-92 extended for querying object data. GemFire supports a subset of OQL.

overflow — Eviction option for eviction controllers. This causes the values of LRU entries to be moved to disk when the region reaches capacity. Writes are performed according to the disk-dirs and disk-write-attributes region attributes. See also persistent-replicate.

partition — The memory in each member that is reserved for a specific partitioned region's use.

partitioned region — A region that manages large volumes of data by partitioning it into manageable chunks and distributing it across multiple machines. Defining partition attributes or setting the region attribute data-policy to `partition` makes the region a partitioned region.

persistent region — A region with the attribute data-policy set to persistent-replicate.

persistent-replicate — A region attribute setting identifying a region as a replicate whose data is persisted to disk. With persistence, all region entry keys and values are stored in an operation log on disk as well as being stored in memory. Also referred to as disk region. Writes are performed according to the disk-dirs and disk-write-attributes region attributes. See also overflow.

pure Java mode — Running GemFire Enterprise without the use of the GemFire native library. GemFire Enterprise can run in this mode with limited capabilities, primarily caching and distribution.

QRegion — The region object representation in a GemFire query. A `QRegion` extends `com.gemstone.gemfire.cache.Region` and `java.util.Collection` so that the single region specification can provide access both to region attributes and to region data collections such as keys and entry values. See also collection.

query string — A fully-formed SQL statement that can be passed to a query engine and executed against a data set. A query string may or may not contain a `SELECT` statement.

race condition — Anomalous behavior caused by the unexpected dependence on the relative timing of events. Race conditions often result from incorrect assumptions about possible ordering of events.

range-index — An XPath index optimized for range-queries with the added index maintenance expense of sorting the set of values. A range index allows faster retrieval of the set of nodes with values in a certain range. See also structure-index and value-index.

region — A logical grouping of data within a cache. Regions are arranged hierarchically and can contain any number of subregions. Regions also can contain data entries (see **entry**). Each region has a set of region attributes governing activities such as expiration, distribution, data loading, events, and capacity control. In addition, a region can have an application-defined user attribute.

region attributes — The class of attributes governing the creation, distribution, and management of a region and its entries.

region data — All of the entries directly contained in the region.

region entry — See entry.

region path — A fully-qualified specification of a region in the cache. The minimum region path is a forward slash (/) followed by a root region name. The region path for a subregion, A1, begins with a forward slash and region A1's root region name followed by the series of

region names required to navigate to region A, with a forward slash delimiter between region names. For example: /A/A1.

remote Resident or running in a cache other than the current member's cache. If a member does not have a data entry in the region in its local cache, it can do a `netSearch` in an attempt to retrieve the entry from the region in a remote cache.

replica A region with data-policy set to replicate or persistent-replicate.

replicate Region data-policy specification indicating to copy all distributed region data into the local cache at region creation time and to keep the local cache consistent with the distributed region data.

role The purpose a member fills in a distributed system, or how a member relates to other members. These optional membership roles specify the circumstances under which a member continues operation after incidents such as network failures. Members can fill one or more roles. Any number of members can be configured to satisfy the same role, and a member can be configured to play any number of roles.

rollback A transactional operation that excludes a transaction's changes from the cache, leaving the cache undisturbed.

root region The topmost region in a cache region hierarchy. There can be multiple root regions in a cache each with its own distinct subregion tree.

scope Region attribute: In non-partitioned regions, a distribution property for data identifying whether it is distributed and, if so, whether distribution acknowledgements are required and whether distributed synchronization is required. A distributed region's cache loader and cache writer (defined in the local cache) can be invoked for operations originating in remote caches. A region that is not distributed has a local scope. See also replicate and data-policy.

Querying: The data context for the part of the query currently under evaluation. The expressions in a `SELECT` statement's `FROM` clause can add to the data that is in scope in the query.

SELECT statement A statement of the form `SELECT` *`projection_list`* `FROM` *`expressions`* `WHERE` *`expressions`* that can be passed to the query engine, parsed, and executed against data in the local cache.

serialization The process of converting an object or object graph to a stream of bytes.

socket The application interface for TCP/IP communications. UDP provides unicast and multicast datagram sockets, while TCP provides server and connection sockets. TCP server sockets are used by server processes to create connection sockets between the server and a client.

SQL **S**tructured **Q**uery **L**anguage.

SSL **S**ecure **S**ocket **L**ayer. A protocol for secure communication between Java VMs.

statistics enabled Region attribute. Specifies whether to collect statistics for the region.

struct A data type that has a fixed number of elements, each of which has a field name and can contain an object value.

structure-index An XPath index that is basically a pre-computed query. Any legal XPath expression can be used. The index maintains lists of all nodes that match the expression used to create it. If a

query is performed that has the same expression as the index then the result is available without XPath evaluation. See also range-index and value-index.

system member A process that has established a connection to a distributed system. This can be an application or a GemFire member such as a locator or cacheserver. The minimal GemFire process configuration is a member that is connected to a distributed system.

TCP The **T**ransmission **C**ontrol **P**rotocol is a part of the internet protocol (IP) suite that provides unicast communications with guaranteed delivery. The TCP protocol is connection-based, meaning that a TCP socket can only be used to send messages between one pair of processes at a time. Compare to UDP.

timeout A behavior associated with synchronization functions. Timeout behavior is exhibited as refusal to wait longer than a specified time for a signal to proceed. See also blocking.

time-to-live The amount of time a region or region entry may remain in the cache without being modified before being expired. Entry modification includes creation, update, and removal. Region modification includes creation, update, or removal of the region or of any of its entries or subregions.

Time-to-live attributes are set at the region level for the region and at the entry level for entries. See also idle timeout and expiration action.

transaction See cache transaction and global transaction.

transactional view The result of a history of transactional operations for a given open transaction.

transport layer The network used to connect the GemFire system members in a GemFire Enterprise system.

TTL See time-to-live.

UDP The **U**ser **D**atagram **P**rotocol is a part of the internet protocol (IP) suite that provides simple, unreliable transmission of datagram messages from one process to another. Reliability must be implemented by applications using UDP. The UDP protocol is connectionless, meaning that the same UDP socket can be used to send or receive messages to or from more than one process. Compare to TCP.

unicast A message sent from one process to another process (point-to-point communications). Both UDP and TCP provide unicast messaging.

URI **U**niform **R**esource **I**dentifier. A unique identifier for abstract or physical resources on the World Wide Web.

user attribute An optional object associated with a region or a data entry where an application can store data about the region or entry. The data is accessed by the application only. GemFire Enterprise does not use these attributes. Compare to region attributes, which are used by GemFire.

value constraint Enforcing a specific entry value type. The `value-constraint` region attribute, when set, constrains the entries in the region to values of the specified object type. Value constraints can be used to provide object typing for region querying and indexing. The value-constraint is only checked in the cache that does the entry put or create operation. When the entry is distributed to other caches, the value constraint is not checked.

value-index An XPath index that operates much as a structure-index does, but that separates the nodes that match the XPath expression into sets mapped by each node's value. This allows further filtering of the nodes to be evaluated in a query by going directly to those with a specific value. See also structure-index and range-index.

view A collection of member identifiers that defines the membership group in JGroups.

VM **V**irtual **M**achine, also referred to as a Java VM.

XML E**X**tensible **M**arkup **L**anguage. An open standard for describing data, XML is a markup language similar to HTML. Both are designed to describe and transform data, but where HTML uses predefined tags, XML allows tags to be defined inside the XML document itself. Thus, virtually any data item can be identified. The XML programmer creates and implements data-appropriate tags whose syntax is defined in a DTD file or an XML schema definition.

XML schema The definition of the structure, content, and semantics used in an XML document. The XML schema is a superset of DTD. Unlike DTD, XML schemas are written in XML syntax, which, although more verbose than DTD, are more descriptive and can have stronger typing. Files containing XML schema definitions generally have the `xsd` extension.

XPath A language that describes a way to locate and process items in Extensible Markup Language (XML) documents by using an addressing syntax based on a path through the document's logical structure or hierarchy.

Index

A

Admin Agent 22
    Configuring 22
administration, overview 25
Adobe Acrobat Reader, installation requirement 16

B

basic administrator tasks 25

C

configuration, attributes 21

D

disk space, installation requirement 16
distributed system, configuring 21
documentation, where installed 20

G

GFMon
    Accessing Information 26
    documentation 20
    installing 15
    Launch 22
    product tree 20

I

installation 15
    requirements 16
    running the installer 18

L

Linux, system requirements 16

O

operating system, installation requirement 16
overview of system administration 25

P

platform installation requirement 16
product directory, contents 20
product documentation, where installed 20
product tree 20

R

RAM installation requirement 16
requirements for installation 16

S

swap space, installation requirement 16
system
    administration, overview 25
    installation requirements 16

T

Technical Support, contacting 15
typographical conventions 9

W

Windows system requirements 16